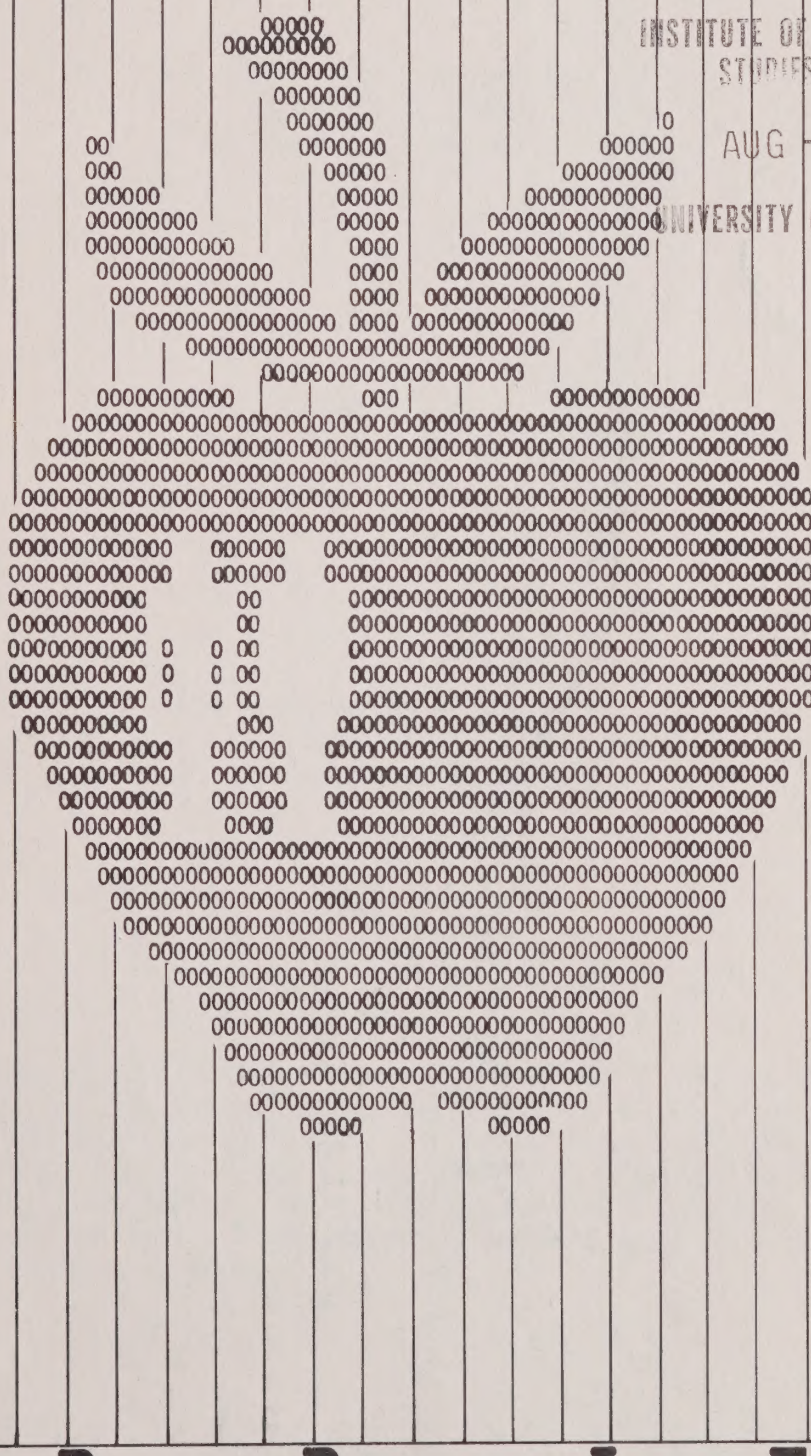
VOLUME II

Small Area Population Projections for Health Planning



ABAG POPULATION PROJECTOR & LABOR ESTIMATOR

SMALL AREA POPULATION PROJECTIONS FOR HEALTH PLANNING

VOLUME II

The Association of Bay Area Governments
Berkeley, California

June 1977

The production of this book was supported by a subcontract with the Health Systems Management Corporation entitled the "Joint Health Planning Project." The Joint Health Planning Program was funded by the Bureau of Health Planning and Resources Development, Health Resources Administration, Department of Health, Education and Welfare under Contract Number HRA 230-75-0069.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ILLUSTRATIONS

PART III

SOFTWARE DOCUMENTATION

"You think far too much of what has happened, instead of what is to become..." Spoken by the ALPHA 60 in "Alphaville", a film by Jean-Luc Godard.

III. SOFTWARE DOCUMENTATION

The first volume contains parts I and II which give the conceptual overview and the complete methodology. This companion volume is for those with computer experience. At one level it tells how to present data to each program needed for the methodology package. At a different level it is aimed at the computer programmer who will install the programs at the health planners computer service bureau. All of the programs are in standard FORTRAN IV and fully tested.

CORE Program: The Cohort Survival Model

Certain forecast data must be prepared for each projection interval. These include forecasts of military in-service and dependent populations, as well as net migration flow size. In addition, a number of age-race-sex-specific ratios are projected by the generators as input to CORE (see figure 1 for general structure):

1. Fertility rates by FERT
2. Survival rates by SURV
3. Labor force participation rates by LFPR
4. Household headship rates by HHEAD
5. Constant data by KGEN, includes school participation rates (for nursery, kindergarten, elementary, high school and college) and group quarters rates (for military barracks, college dormitories, and other group quarters).

Required Input

CARD	COLUMNS	DESCRIPTION
1		RUN CARD (I2, 1X, I1, 1X, I4, 5X, I1, 1X, I4)
	1-2	Output amount (1=minimum, 2=medium, 3=maximum)
	4	Number of races (1 or 2)
	6-9	Base year
	20	"1" if graphics are desired, otherwise blank
	22-25	End year for forecast
2		FIRST COMMENT CARD (20A4)
	1-80	Comment on run
3		SECOND COMMENT CARD
		Same as Card No. 2

CARD	COLUMNS	DESCRIPTION
4		MISCELLANEOUS INTERVAL DATA (5F10.0)
	1-10	Net migration flow for interval
	11-20	Unemployment rate-fraction of labor force unemployed (decimal must be punched)
	21-30	Jobs per employee-average number of jobs held by each worker (decimal must be punched)
	31-40	Vacancy rate-fraction of total housing units which are unoccupied (demical must be punched)
	41-50	Fraction of total school enrollment attending public schools (decimal must be punched)
5,6, etc.		ONE INTERVAL CARD REQUIRED FOR EACH PROJECTION INTERVAL, (e.g., 1970-2000 requires 6 such cards)

Output

Printed output will be produced according to the print code on the Run Card. Age-sex pyramid are additional options. Distribution tables include numeric and percent distributions arranged by age groups, sex, and total. Computed totals are rounded to the nearest whole number. Because of rounding, the total may not equal the sum of the parts. Percents are rounded to the nearest tenth. Summary Table components are given to the nearest hundred.

Output Print Control

Print code punched in Column 2 of Run Card:

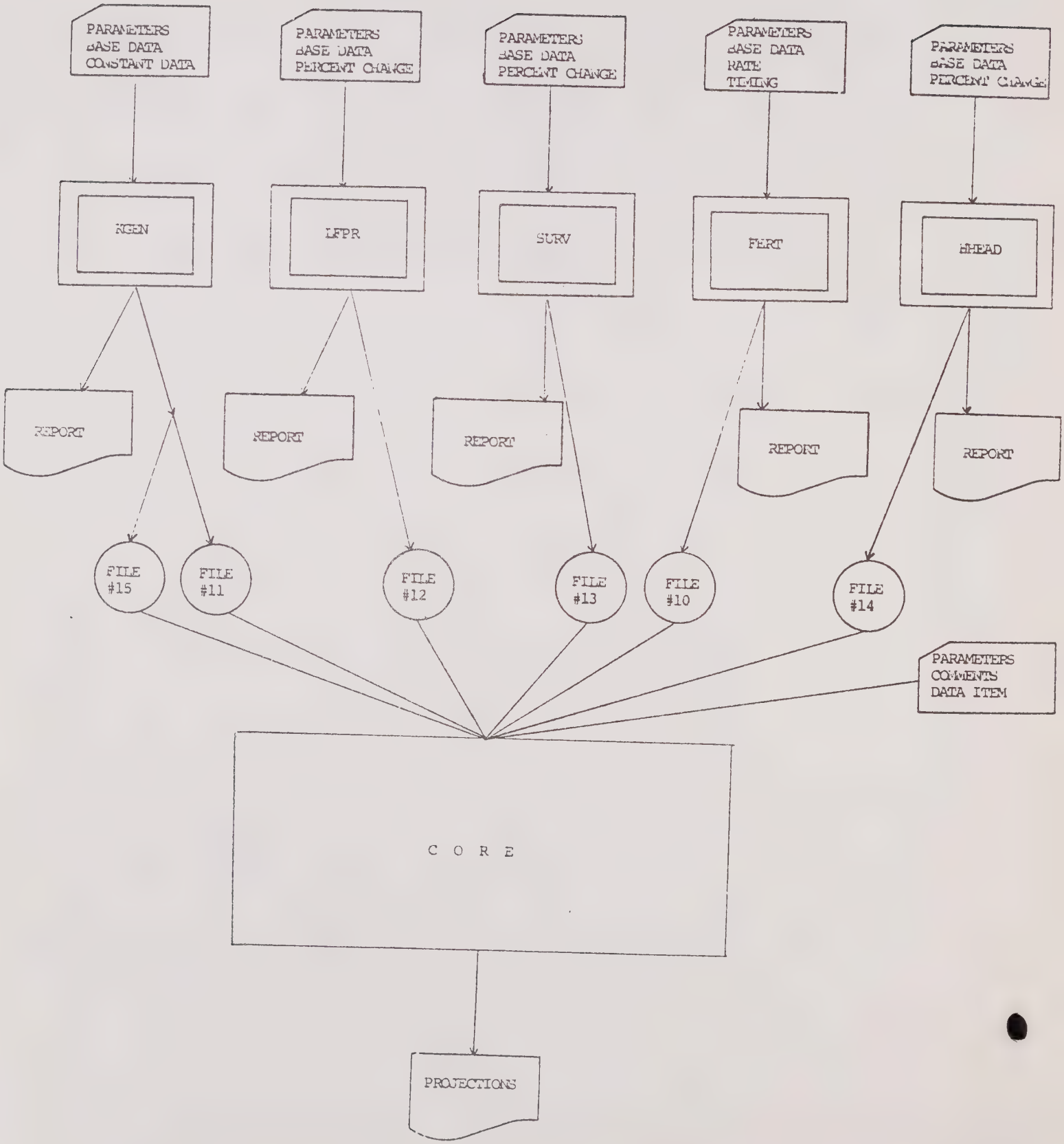
1 -Minimum output (total population and employment projections only)

2 -Medium output (all data except cumulative figures)

3 -Maximum output (all data plus cumulative figures). Code 3 will print out the age-sex distributions of the resident and total populations each time they are incremented.

FIGURE 1

APPLE SYSTEM



Description of Output

The following is a description of the output of the program that will be printed with the print codes indicated.

	PRINT CODES
1. Run Description -	
a. Listing of run parameters	1,2,3
b. Card image printout of two comment cards	1,2,3
c. Percent births male, white and nonwhite	2,3
d. Percent race-age-sex distribution of migrants	2,3
2. Base year population	
a. Resident population by age and sex - white, nonwhite and total	2,3
b. Total military population by age and sex - white, nonwhite and total	2,3
c. Total population by age and sex - white, nonwhite, and total	1,2,3
3. Description of interval data	
a. Survival rates by age, race, and sex	2,3
b. Births and survived residents by age and sex at end of interval - white, nonwhite, and total	2,3
c. Retirement migrants by age and sex - white, nonwhite, and total	2,3
d. Military dependent population by age and sex - white, nonwhite, and total	2,3
e. School participation rates (nursery, kindergarten, elementary, high school, and college) by race-age-sex	2,3
f. Percent group quarters (military barracks, college dormitories, and other) by race-age-sex	2,3
g. Household headship rates by race-age-sex	2,3
h. Details of population change	2,3

	PRINT CODES
i. Employment projection total	1,2,3
j. Employment projection by industry, sorted by basic, household-serving and business-serving categories. Includes unemployment rate assumed.	2,3
k. Migrants by age and sex - white, nonwhite, and total	2,3
l. Military in-service population by age and sex - white, nonwhite, and total	2,3
m. Labor force participation rates by race-age-sex	2,3
4. Description of data produced at end of each interval	
a. School enrollment (nursery, kindergarten, elementary, high school, college, and total) by age and sex - white, nonwhite, and total	2,3
b. Group quarters population (military barracks, college dormitories, other, and total) by age and sex - white, nonwhite, and total	2,3
c. Household population by age and sex - white, nonwhite, and total	2,3
d. Heads of households by age and sex - white, nonwhite, and total	2,3
e. Military in-service population by age and sex - white, nonwhite, and total	2,3
f. Total population by age and sex - white, nonwhite, and total	1,2,3
g. Total population change over interval by age and sex - white, nonwhite, and total	1,2,3
h. Labor force by age and sex - white, nonwhite, and total	2,3
i. Resident population by age and sex - white, nonwhite, and total	2,3
5. Description of data produced at end of projection	
a. Total population change over projection period by age and sex - white, nonwhite, and total	1,2,3

PRINT CODES

b. Summary of components of population growth. Includes total population change, resident population change (natural increase + migration) and military-related population change for each interval of the projection. 1,2,3

c. Projection Summary. Includes total population, employment, labor force, school enrollment, household population, household heads, and group quarters population for base year and each end-of-interval year 1,2,3

Description of Graphics

An age-sex pyramid will be produced for each projection interval if "1" is punched in column 15 of the Run Card. The pyramids are subdivided by male/female in 5 year age groups. The pyramid subroutine determines the appropriate scale to be used for each table, the scales ranging from a minimum of 1=100 to a maximum of 1=30,000. Pyramids are printed for the following tables:

1. Base year resident population
2. Base year total military population
3. Base year total population
4. Births and survived population at the end of each interval
5. Retirement-related migrant population for each interval
6. Military dependent population at the end of each interval
7. Employment-related migrant population for each interval
8. Military in-service population at the end of each interval
9. Total population at the end of each interval
10. Resident population at the end of each interval.

In addition, composite pyramids are produced for end-of-interval total population and total population at the end of the previous interval. Composite pyramids are also printed for end-of-projection total population and base year total population.

Use of the option RACES=2 will result in the printing of pyramids for white and nonwhite as well as total for each of the tables mentioned. Figures 2 and 3 contain examples of age-sex pyramids. Table 1 is a cost comparison.

FIGURE 2

EXAMPLE OF AGE-SEX PYRAMID FOR TOTAL POPULATION

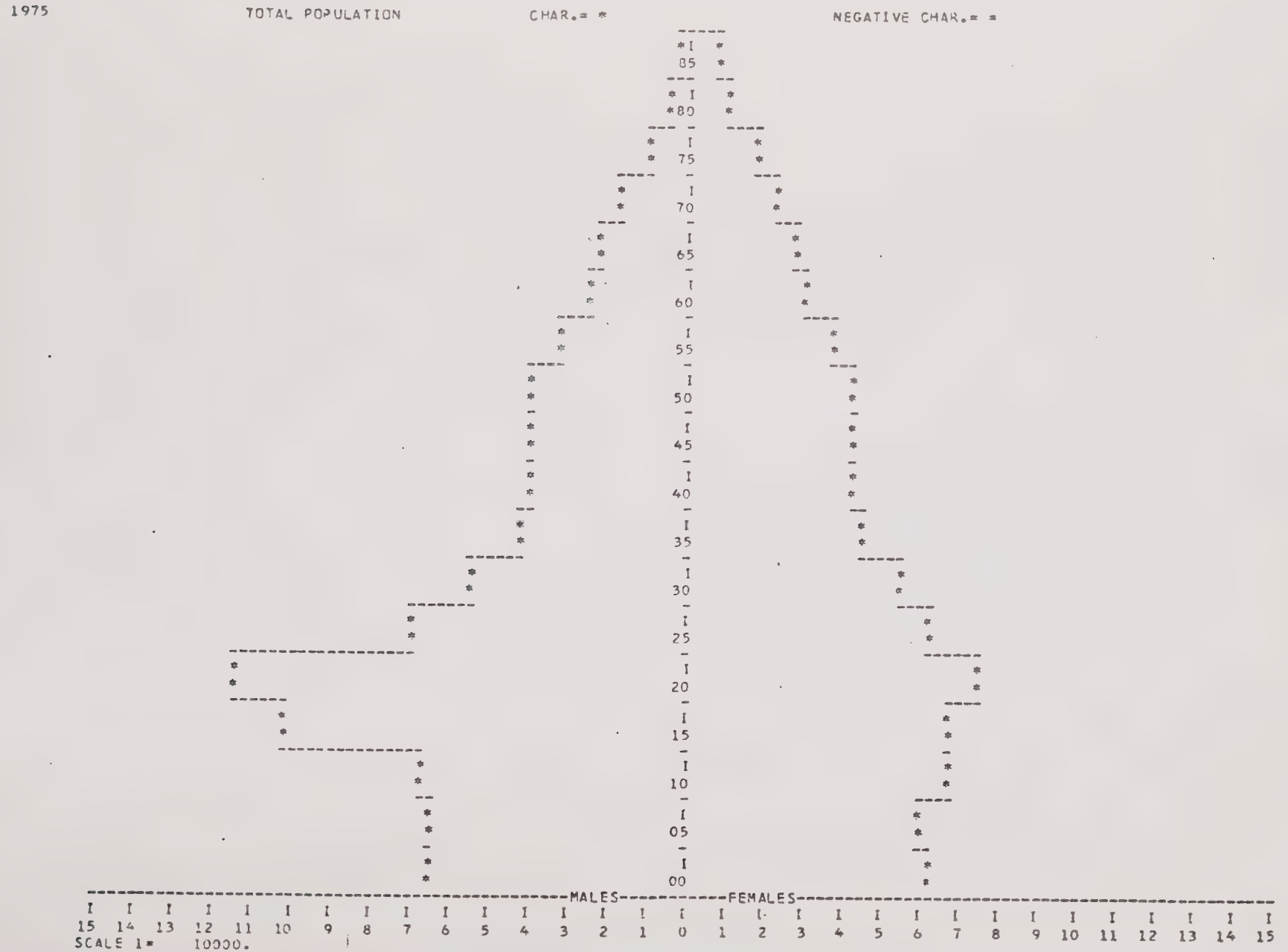


FIGURE 3

EXAMPLE OF AGE-SEX PYRAMID COMPARING BASE AND FORECAST POPULATIONS

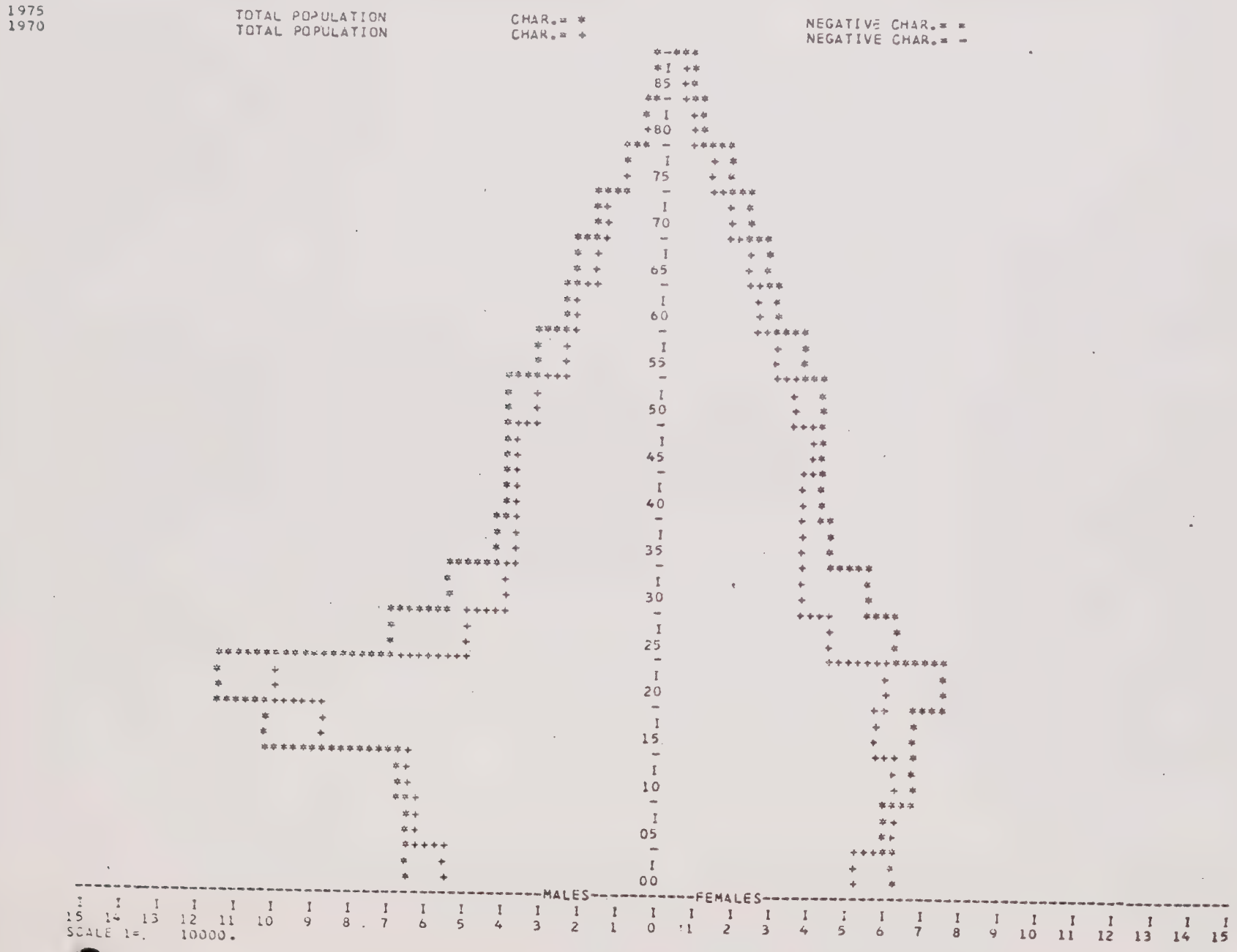


TABLE 1

IMPACT OF VARIOUS LEVELS OF OUTPUT
AND GENERATION OF PYRAMIDS

		OUTPUT LEVEL REQUESTED		
		1	2	3
PYRAMIDS	YES	5.06	18.48	18.58
	NO	1.16	4.40	5.21

The sample runs represent the extreme low estimates of the fertility, survival, and migration.

```
C*     THE ORIGINAL APPLE - CASTRO,LANKFORD,SHYER
C*     BASED ON THE INTERACTIVE POPULATION AND EMPLOY-
C*     MENT FORECASTING MODEL
C*     MODIFIED AND ENHANCED BY PHIL LANKFORD AND
C*     NANCY SHYER
C*
C*     * POPULATION FORECASTING
C*
C*     * VARIABLE NAMING CONVENTIONS
C*         DIMENSION 20 IS TOTAL OF 1 - 19
C*                   X1 - W=WHITE,N=NONWHITE,T=TOTAL,H=HOLD-AREA
C*                   X2 - M=MALE,F=FEMALE,B=BOTH
C*                   X3 - 0=BASE YEAR,I=CURRENT VALUE,C=CALCULATED VALUE
C*                        P=PROPORTION,R=RATE
C*         POPXXX = POPULATION
C*         TOTXXX = TOTAL POPULATION
C*         RESXXX = RESIDENT POPULATION
C*         SPCXXX = SPECIAL POPULATION (MILITARY)
C*         MIGXXX = MIGRANTS
C*         FERXXX = FERTILITY - DIMENSION 11 IS SUM OF 5-10
C*         LABXXX = LABOR FORCE,   PARTICIPANTS
C*         SCHXXP(I,7) = SCHOOL PARTICIPATION RATES
C*         SCHXXE(I,7) = SCHOOL ENROLLMENT
C*                I=1    NURSERY
C*                  2    KINDERGARTEN
C*                  3    ELEMENTARY
C*                  4    HIGH SCHOOL
C*                  5    COLLEGE
C*                  6    TOTAL (ENROLLMENT ONLY)
C*         GQXXXX = GROUP QUARTERS
C*         HHXXXX = HOUSEHOLD HEADS
C*         BIRXXX = BIRTHS
C*         SURXXX = SURVIVAL RATES
C*         TEMXXX = TOTAL EMPLOYMENT
C*         UNMXXX = UNEMPLOYMENT RATE
C*         PCTX   = PERCENTAGE              CELL 20 = TOTAL OF 1-19
C*         OUTXXX = AMOUNT OF PRINTED OUTPUT FLAG
C*                  VALUE  3 = MAXIMUM
C*                         2 = MEDIUM
C*                         1 = MINIMUM
C*         RACES  = MORE THAN 1 RACE
C*                  VALUE  1 = NO SUBDIVISION BY RACE
C*                         2 = SUBDIVISION BY RACE
C*         RUN    = RUN NUMBER
C*         YEAR   =  YEAR FOR FORECAST
C*         LASTYR = LAST YEAR, YEAR OF LAST FORECAST
C*         BASEYR = BASE YEAR FOR RUN
C*         BYRED = BASE YEAR FOR EMPLOYMENT DATA
C*         INT    = INTERVAL NUMBER (0 = BASE YEAR)
C*               TABLE = TABLE NUMBER
C*                       VALUE 0 = TOTAL
C*                             1 = RESIDENT
C*                             2 = SPECIAL (MILITARY)
C*                             3 = MIGRANT
C*                             4 = CUMULATIVE TOTAL
C*                             5 = CUMULATIVE RESIDENT
C*                             6 = BIRTHS AND SURVIVED
C*                             7 = MILITARY DEPENDENT
C*                             8 = MILITARY IN-SERVICE
C*                             9 = EMPLOYMENT RELATED MIGRANT
C*                            10= RETIREMENT RELATED MIGRANT
```

```
C*                                11= LABOR FORCE
C*                                12= SCHOOL ENROLLMENT
C*                                14= GROUP QUARTERS
C*        CYCLE  = RUN CYCLE NUMBER
C*               VALUE 1 = ANOTHER INTERVAL FOLLOWS, SAME FORECAST
C*                     2 = LAST INTERVAL OF FORECAST
C*        RNAMES = RACE NAMES
C*        TNAMES = TABLE NAMES
C*        PYRMDS = PYRAMID ? BAR GRAPH CONTROL
C*                 1 = YES
C*                 0 = NO
C*
C*
C*     C** INDICATES THE CHANGES MADE AS OF 9/1/75
       REAL    LABTBC,LABXXX,LABWMP,LABWFP,LABNMP,LABNFP,NGRI,INTYRS
       REAL JPEMP,KBASIC,KLFPR,LABNFC,LABNMC,LABWFC,LABWMC,LAST
       INTEGER GQN1(10),GQN2(10),ENDYR
       REAL    MIGTBC,MULTW,MULTX,MULTY,MULTZ,MIGTOT

       REAL*8 LIT(30),SAGE(7),GA(10)
       COMMON /PER/PCTM(20),PCTF(20),PCTT(20)
       COMMON       /PRT/     OUTXXX,SPECL,YEAR,LASTYR,RACES,PAGE,PYRMDS
       COMMON /EMP/TCR,UNMTBP,LABTBC,LABXXX,BEI,HSI,BSI,AR,NGRI
       COMMON/LAB/LABWMP(20),LABWFP(20),LABNMP(20),LABNFP(20)
       COMMON /TOT/TOTWMC(20),TOTWFC(20),TOTNMC(20),TOTNFC(20),
      1            TOTBMC(20),TOTBFC(20)
       COMMON /FER/FERWFR(10),FERWFC(10),FERNFR(10),FERNFC(10)
       COMMON /RES/RESWMC(20),RESWFC(20),RESNMC(20),RESNFC(20),
      1            RESBMC(20),RESBFC(20)
       REAL*8 IN1(100),IN2(100),IN3(100),IN4(100),IN5(100),B(100)
       REAL       TOTWMO(20),TOTWFO(20),TOTNMO(20),TOTNFO(20),
      *           TOTBMO(20),TOTBFO(20),TOTBMI(20),TOTBFI(20),
      *           TOTWMI(20),TOTWFI(20),TOTNMI(20),TOTNFI(20),
      *           RESWMI(20),RESWFI(20),RESNMI(20),RESNFI(20),
      *           SPCWMI(20),SPCWFI(20),SPCNMI(20),SPCNFI(20),
      *           MIGWMP(20),MIGWFP(20),MIGNMP(20),MIGNFP(20),
      *           MIGWMC(20),MIGWFC(20),MIGNMC(20),MIGNFC(20),
      *           SURWMP(20),SURWFP(20),SURNMP(20),SURNFP(20),
      *           POPHMC(20),POPHFC(20)
       REAL                            DWTOT(7),TOTAL4(7),DWFDS(7,10),
      *CODE(10),CC1(19),CC2(19),
      *           DWBD(10),DWFD(10),BPERM(100),DWBDP(10),ELAST(100),
      *           B2(100),B3(100),BE(100),E(100)
       REAL       SCHWMP(5,7),SCHWFP(5,7),SCHNMP(5,7),SCHNFP(5,7),
      *           SCHWME(6,8),SCHWFE(6,8),SCHNME(6,8),SCHNFE(6,8),
      *           SCHMTE(6,8),SCHFTE(6,8),NGR(100,7)
       REAL       TPLSWM(20),TPLSWF(20),TPLSNM(20),TPLSNF(20),
      *           LFWM(20),LFWF(20),LFNM(20),LFNF(20),LFTM(20),LFTF(20),
      *           GQMWMP(10),GQMNMP(10),GQCWMP(10),GQCWFP(10),
      *           GQCNMP(10),GQCNFP(10),GQOWMP(10),GQOWFP(10),
      *           GQONMP(10),GQONFP(10),GQMWM(11),GQMNM(11),
      *           GQCDWM(11),GQCDWF(11),GQCDNM(11),GQCDNF(11),
      *           GQOWM(11),GQOWF(11),GQONM(11),GQONF(11),
      *           GQMWF(11),GQMNF(11),GQMWFP(10),GQMNFP(10)
       REAL       GQWMT(11),GQWFT(11),GQNMT(11),GQNFT(11),
      *           GQMT(11),GQFT(11),GQCMT(11),GQCFT(11),
      *           GQMMT(11),GQMFT(11),GQCDMT(11),GQCDFT(11),
      *           HHWMP(10),HHWFP(10),HHNMP(10),HHNFP(10),
      *           HHWMC(11),HHWFC(11),HHNMC(11),HHNFC(11),
      *           HHMTC(11),HHFTC(11)
       REAL       HHPWM(11),HHPWF(11),HHPNM(11),HHPNF(11),
```

```
*           MHPMTC(11),MHPFTC(11),GQP(11),
*           TOTWMT(20),TOTWFT(20),TOTNMT(20),TOTNFT(20),
*           MHPWMP(20),MHPWFP(20),MHPNMP(20),MHPNFP(20),
*           CEWM(20),CEWF(20),CENM(20),CENF(20),
*           CPWM(20),CPWF(20),CPNM(20),CPNF(20),
*           KGQ(3,4),TS(3,6)
 INTEGER NUMC2,NN1,NN2,IFMT(5),JFMT(7),KFMT(7),ID,          MYEARS,SW
*,TZ(101),
*         AL(10),K,J,L,JJ,MM,JK,NI,NN,I,M,M1,M2,M3,NUMYR,NYR,
*         C1(100),C2(100),SIC(100),BYR,YRCT,YEAR,PAGE,OUTXXX,
*         RACES,RACE,SEX,AGE,CYCLE,BASEYR,INDEXM,INDEXF,YEAR2,
*         PYRMDS,BYRED,N,INT,TYPE,LASTYR,TABLE,     AGE1,AGE2,LYREMP
 INTEGER SUMMYR(11),TOTPOP(30),RESPOP(30),MISPOP(30),DEPPOP(30),
*         TOTEMP(30),BASEMP(30),HSEMP(30),BSEMP(30),LABFOR(30),
*         SCHENR(30),HHH(30),SCHNUR(30),SCHKIN(30),SCHEL(30),
*         SCHHS(30),SCHCOL(30),TOTGQ(30),GQMB(30),GQCD(30),GQO(30),
*         HHPOP(30),CONEMP(30),PSEMP(30),TOTCH(30),RESCH(30),
*         BIRTHS(30),DEATHS(30),NATI(30),MIGT(30),MIGE(30),
*         MIGR(30),MILREL(30)
 DATA MIGWMC,MIGWFC,MIGNMC,MIGNFC/80*0./
 DATA     GQN1     /1,4,5,6,7,8,10,12,14,16/,
*         GQN2     /3,4,5,6,7,9,11,13,15,18/
 DATA CPWM,CPWF,CPNM,CPNF/80*0./
 DATA CEWM,CENM,CEWF,CENF/80*0./
 DATA TS/4HNURS,4HERY ,4H     ,4HKIND,4HERGA,4HRTEN,
*          4HELEM,4HENTA,4HRY  ,4HHIGH,4H SCH,4HOOL ,
*          4HCOLL,4HEGE ,4H    ,4HTOTA,4HL   ,4H    /
 DATA KGQ/4HMILI,4HTARY,4H    ,
*          4HCOLL,4HEGE ,4HDORM,
*          4HOTHE,4HR   ,4H    ,
*          4H    ,4H    ,4H    /
 DATA LIT/8HAGRICULT,8HURE      ,8H         ,
*          8HMINERAL ,8HEXTRACTI,8HON       ,
*          8HCONTRACT,8H CONSTRU,8HCTION    ,
*          8HMANUFACT,8HURING   ,8H         ,
*          8HTRANS,CO,8HMM,PUBLI,8HC UTIL   ,
*          8HWHOLESAL,8HE ? RETA,8HIL TRADE,
*          8HFIN,INSU,8HR,REAL E,8HSTATE    ,
*          8HSERVICE ,8H        ,8H         ,
*          8HGOVERNME,8HNT      ,8H         ,
*          8HOTHER NO,8HN AGRICU,8HLTURE    /
 DATA SAGE/    5H 0- 4,5H 5- 9,5H10-14,5H15-19,
*              5H20-24,5H25-29,5H30-34/
 DATA GA/5H 0-14,5H15-19,5H20-24,5H25-29,5H30-34,
*            5H35-44,5H45-54,5H55-64,5H65-74,5H75+  /
 DATA IFMT/4H(26X,4H,    ,1H1,4H(1H*, 4H))   /
 DATA JFMT/4H(2X,,4H3A8,,4H1H*,,0,4H(1H+,4H),1H,2H*) /
 DATA KFMT/4H(23X,4H,I4,,4H2X,1,4HH), ,0,4HX,1H,4HX)  /
 DATA CODE/1HA,1HB,1HC,1HD,1HE,1HF,1HG,1HH,1HI,1HJ/
 DATA TZ/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9,2H10,
*        2H11,2H12,2H13,2H14,2H15,2H16,2H17,2H18,2H19,2H20,
*        2H21,2H22,2H23,2H24,2H25,2H26,2H27,2H28,2H29,2H30,
*        2H31,2H32,2H33,2H34,2H35,2H36,2H37,2H38,2H39,2H40,
*        2H41,2H42,2H43,2H44,2H45,2H46,2H47,2H48,2H49,2H50,
*        2H51,2H52,2H53,2H54,2H55,2H56,2H57,2H58,2H59,2H60,
*        2H61,2H62,2H63,2H64,2H65,2H66,2H67,2H68,2H69,2H70,
*        2H71,2H72,2H73,2H74,2H75,2H76,2H77,2H78,2H79,2H80,
*        2H81,2H82,2H83,2H84,2H85,2H86,2H87,2H88,2H89,2H90,
*        2H91,2H92,2H93,2H94,2H95,2H96,2H97,2H98,2H99,3H100/
 DATA GQMWF,GQMNF/11*0.,11*0./
 DATA GQMWFP,GQMNFP/10*0.,10*0./
```

```
      DATA GQP/11*1./
      PAGE = 1
C**   TRANSFER CONTROL TO THE REPORT TITLING ROUTINE,TITLEP
      CALL TITLEP
C*    READ RUN CARD
  100 READ(5,901,END=7532)     OUTXXX,RACES,BASEYR,BYRED,PYRMDS,ENDYR
      MIGIN = 1
  901 FORMAT(212,2I5,4X,I2,I5)
      CALL PRINTT(21)
      WRITE(6,9001)     OUTXXX,BASEYR,BYRED,RACES,PYRMDS,MIGIN,ENDYR
 9001 FORMAT(1HO,2X,'RUN PARAMETERS',/,5X,9X,              /,
     * 5X,'OUTXXX =',I2,/,5X,'BASE YEAR = ',I4,/,
     * 5X,'BASE YEAR FOR EMPLOYMENT DATA = ',I4,/,
     * 5X,'RACES =',I2,/,5X,'PYRAMIDS =',I2,/,5X,'MIGRATION=',I2,/,
     * 5X,'END YEAR OF FORECAST  = ', I5)
      READ(5,903) CC1,CC2
  903 FORMAT(20A4/20A4)
      WRITE(6,9002) CC1,CC2
 9002 FORMAT(/,1X,19A4,/,1X,19A4)
      READ(11,904) DEPPOP(1),LABFOR(1),SCHNUR(1),SCHKIN(1),SCHEL(1),
     *          SCHHS(1),SCHCOL(1),HHH(1),GQMB(1),GQCD(1),GQO(1)
  904 FORMAT(8I7,3I6)
      SCHENR(1) = SCHNUR(1)+SCHKIN(1)+SCHEL(1)+SCHHS(1)+SCHCOL(1)
      TOTGQ(1) = GQMB(1) + GQCD(1) + GQO(1)
      YEAR = BASEYR
      INT = 0
      SUMMYR(1) = BASEYR
      DO 777 I=2,11
      SUMMYR(I) = SUMMYR(I-1) + 5
  777 CONTINUE
C*
C*    READ   BIRTHS MALE
C*     READ MIGRANT DISTRIBUTION  (LABOR FORCE RELATED MIGRANTS)
C*
      READ(11,914) BIRWMP,BIRNMP
  914 FORMAT(F3.3,1X,F3.3)
      IF(OUTXXX.EQ.1) GO TO 916
      WRITE(6,915) BIRWMP,BIRNMP
  915 FORMAT(1HO,2X,'  BIRTHS MALE WHITE = ',F5.4,/,
     *          3X,'  BIRTHS MALE NONWHITE = ',F5.4)
  916 CONTINUE
      READ(11,917)  MIGWMP,MIGWFP,
     1              MIGNMP,MIGNFP
  917 FORMAT (13F6.4/7F6.4)
      IF(OUTXXX.EQ.1) GO TO 9016
      WRITE (6,9010)
 9010 FORMAT(//' PERCENT DISTRIBUTION OF MIGRANTS(LABOR FORCE RELATED)')
      WRITE (6,9015)
 9015 FORMAT ( /,'  AGE      WHITE      WHITE      NONWHITE  NONWHITE',/,
     1          ' GROUP     MALE       FEMALE     MALE      FEMALE  ',/ )
      AGE1 = -5
      DO 360 N=1,18
      AGE1 = AGE1 +5
      AGE2 = AGE1 +4
      IF (AGE2 .EQ. 89) AGE2 = 999
  360 WRITE (6,9020) AGE1,AGE2,MIGWMP(N),MIGWFP(N),MIGNMP(N),MIGNFP(N)
 9020 FORMAT (' ',I2,'-',I2,3X,4(F6.4,4X))
 9016 CONTINUE
C*
C*    READ BASE YEAR POPULATIONS, WHITE, NONWHITE, SPECIAL
C*
```

```
      READ(11,910)  (TOTWMI(N),N=1,19),(TOTWFI(N),N=1,19),
     1              (TOTNMI(N),N=1,19),(TOTNFI(N),N=1,19),
     2              (SPCWMI(N),N=1,19),(SPCWFI(N),N=1,19),
     3              (SPCNMI(N),N=1,19),(SPCNFI(N),N=1,19)
  910 FORMAT (9F7.0,/,10F7.0)
C*
C***** READ EMPLOYMENT BASE YR DATA *****
C*    NUMBER OF INDUSTRIES, AEROSPACE, DWELLING UNITS,
C*    SCHOOL ENROLLMENT, STATE COLLEGE ENROLLMENT, POPULATION
C*
      READ(11,9601) NI,BAR,BDU,BENR,BSCE,EPOP
C**   THIS IS A DESPERATE ATTEMPT AT DEBUGGING
      WRITE(6,9601) NI,BAR,BDU,BENR,BSCE,EPOP
 9601 FORMAT(I5,5F10.0)
      DUI = BDU
      TBBEI=0.0
      TBHSI=0.0
      TBBSI=0.0
      TBCON=0.0
      TBPS=0.0
C**   IF(OUTXXX.EQ.1) GO TO 9620
C**   CALL PRINTT(19)
C**   WRITE(6,9603)
C** 9603 FORMAT(36X,'EMPLOYMENT FORECASTS',/,38X,'BASE YEAR DATA',//,
C**  *44X,'       BASE YR',19X,'NATIONAL GROWTH RATE FOR INTERVAL -',/,
C**  *' C1 C2 SIC',9X,'INDUSTRY NAME',12X,'BETA    EMPLOY',10X,
C**  *'0       1       2       3       4       5       6'/)
C** 9620 CONTINUE
C*
C*    READ INDUSTRY TYPE, CLASS, CODE, NAME, REGRESSION PARAMETER,
C*    BASE YEAR VALUE, NATIONAL ANNUAL RATES OF GROWTH FOR BASIC
C* C1(I)=1 FOR BASIC INDUSTRY
C* C1(I)=2 FOR HOUSEHOLD SERVING INDUSTRY
C* C1(I)=3 FOR BUSINESS SERVING INDUSTRY
C* C1(I)=4 FOR CONSTRUCTION
C* C1(I)=5 FOR LOCAL PUBLIC SCHOOLS
C* C1(I)=6 FOR STATE COLLEGE EMPLOYMENT
C*
      IF (MIGIN .GT.0) GO TO 2222
      DO 10 I=1,NI
      READ(5,9602)C1(I),C2(I),SIC(I),IN1(I),IN2(I),IN3(I),IN4(I),IN5(I),
     1B(I),BE(I),(NGR(I,J),J=1,7)
 9602 FORMAT(I1,A1,A3,4A6,A4,F8.0,F6.1,7F4.3)
      IF(OUTXXX.EQ.1) GO TO 9621
C* PRINT BASE YEAR DATA
C**   WRITE(6,9604)C1(I),C2(I),SIC(I),IN1(I),IN2(I),IN3(I),IN4(I),
C**  *              IN5(I),B(I),BE(I),(NGR(I,J),J=1,7)
 9604 FORMAT(I3,A3,A4,2X,4A6,A4,F9.4,F7.1,6X,7(3X,F5.3))
 9621 CONTINUE
      BE(I) = BE(I)*1000.
      NN=C1(I)
C* CALCULATE BASE YEAR SUBTOTALS
      GO TO (21,22,23,25,26,21),NN
   21 TBBEI=TBBEI+BE(I)
      GO TO 10
   22 TBHSI=TBHSI+BE(I)
      GO TO 10
   23 TBBSI=TBBSI+BE(I)
      GO TO 10
   25 TBCON=TBCON+BE(I)
      GO TO 10
```

```
   26 TBPS=TBPS+BE(I)
   10 CONTINUE
 2222 BARTH = BAR/1000.
C**   WRITE(6,9605) BARTH
 9605 FORMAT(12X,'OTHER EMPLOYMENT',21X,F7.1)
C*
      BASEMP(1) = TBBEI + BAR
      HSEMP(1) = TBHSI
      BSEMP(1) = TBBSI
      CONEMP(1) = TBCON
      PSEMP(1) = TBPS
      TOTEMP(1)=BASEMP(1)+HSEMP(1)+BSEMP(1)+CONEMP(1)+PSEMP(1)
      DO 670 NN=1,10
  670 DWBD(NN)=0.0
C*    STORING BASE DATA SUBTOTALS BY C2 CODE
      DO 676 I=1,NI
      IF(NI .EQ. 0) GO TO    676
      DO 675 NN=1,10
      IF(C2(I).EQ.CODE (NN))GO TO 674
675   CONTINUE
      GO TO 676
674   C2(I)=NN
      DWBD(NN)=DWBD(NN)+BE(I)
676   CONTINUE
      DWBD(4)=DWBD(4)+BAR
C*    STORE ORIGINAL BASE DATA FIRST TIME THRU
      IF(YEAR.NE.BASEYR)GO TO 677
      TOTAL1=0.0
      ESIC = 0.0
      DO 678 NN=1,10
      DWFDS(1,NN)=DWBD(NN)
      TOTAL1=TOTAL1+DWBD(NN)
678   DWBDP(NN)=DWBD(NN)
      TOTAL4(1)=TOTAL1
      DO 680 I=1,NI
      IF(NI .EQ. 0) GO TO    680
      IF(SIC(I).EQ.SIC(I+1)) GO TO 679
      BPERM(I) = ESIC + BE(I)
      ELAST(I) = BPERM(I)
      ESIC = 0.0
      GO TO 680
  679 ESIC = ESIC + BE(I)
  680 CONTINUE
      TOTAL2=TOTAL1
      PBAR=BAR
677   CONTINUE
C*
C*    DERIVE RESIDENT POPULATION
C*
  125 DO 150 N=1,19
      RESWMI(N) = TOTWMI(N) - SPCWMI(N)
      RESWFI(N) = TOTWFI(N) - SPCWFI(N)
      RESNMI(N) = TOTNMI(N) - SPCNMI(N)
      RESNFI(N) = TOTNFI(N) - SPCNFI(N)
  150 CONTINUE
C*
C*
C*    PRINT RESIDENT,SPECIAL, AND TOTAL BASE POPULATION
C*
      CALL TOTAL(RESWMI,RESWFI,RESNMI,RESNFI,POPHMC,POPHFC,1)
      RESPOP(1) = POPHMC(20) + POPHFC(20)
```

```
      CALL TOTAL(SPCWMI,SPCWFI,SPCNMI,SPCNFI,POPHMC,POPHFC,2)
      MISPOP(1) = POPHMC(20) + POPHFC(20) - DEPPOP(1)
      CALL TOTAL(TOTWMI,TOTWFI,TOTNMI,TOTNFI,POPHMC,POPHFC,0)
      TOTPOP(1) = POPHMC(20) + POPHFC(20)
      EPOP = TOTPOP(1)
C*
C*    SAVE BASE POPULATION DATA
C*
      DO 200 N=1,20
      TOTWMO(N) = TOTWMI(N)
      TOTWFO(N) = TOTWFI(N)
      TOTNMO(N) = TOTNMI(N)
      TOTNFO(N) = TOTNFI(N)
      TOTBMO(N) = TOTWMO(N) + TOTNMO(N)
      TOTBFO(N) = TOTWFO(N) + TOTNFO(N)
      TOTBMI(N) = TOTBMO(N)
      TOTBFI(N) = TOTBFO(N)
  200 CONTINUE
      OWTOT(1) = TOTBMO(20) + TOTBFO(20)
      K=1
      JK=1
C*
C*          **********   START OF INTERVAL ITERATION   **********
C*
C*
  400 LASTYR = YEAR
      INT = INT + 1
      LYREMP = LASTYR
      IF(YEAR.EQ.BASEYR) LYREMP = EYRED
      NUMYR = INT + 1
      IF(YEAR.EQ.BASEYR) GO TO 401
      CALL TOTAL(RESWMI,RESWFI,RESNMI,RESNFI,POPHMC,POPHFC,1)
      RESPOP(INT) = POPHMC(20) + POPHFC(20)
      RESCH(INT-1) = RESPOP(INT) - RESPOP(INT-1)
  401 YEAR = YEAR + 5
      INTYRS = YEAR - LYREMP
C*
C*    READ SURVIVAL, FERTILITY RATES, CALCULATE SURVIVAL, FERTILITY
C*
      READ(13,920)          (SURWMP(N),N=1,19),(SURWFP(N),N=1,19),
     I                (SURNMP(N),N=1,19),(SURNFP(N),N=1,19)
  920 FORMAT (14F5.5/5F5.5)
C*
C*    PRINT SURVIVAL RATES
C*
      IF (OUTXXX .EQ. 2 .AND. INT .NE. 1) GO TO 550
      IF (OUTXXX .EQ. 1) GO TO 550
      CALL PRINTT(22)
      WRITE (6,9040) LASTYR,YEAR
 9040 FORMAT (  ' SURVIVAL RATES ',I4,'-',I4)
      WRITE (6,9015)
      WRITE (6,9050) SURWMP(1),SURWFP(1),SURNMP(1),SURNFP(1)
 9050 FORMAT (' BIRTH    ', 4(F6.5,4X))
      AGE1 = -5
      DO 500  N=2,19
      AGE1 = AGE1 + 5
      AGE2 = AGE1 + 4
      IF (AGE2 .EQ. 89) AGE2 = 999
  500 WRITE (6,9060) AGE1,AGE2,SURWMP(N),SURWFP(N),SURNMP(N),SURNFP(N)
 9060 FORMAT (' ',I2,'-',I2,3X,4(F6.5,4X))
  550 READ(10,930)  (FERWFR(N),N=4,09),
```

```
     1               (FERNFR(N),N=4,09)
 930  FORMAT (6F4.1)
C*
      CALL SURV(RESWM1,RESWF1,RESWMC,RESWFC,SURWMP,SURWFP,
     1          FERWFR,FERWFC,BIRWMP)
      CALL SURV(RESNM1,RESNF1,RESNMC,RESNFC,SURNMP,SURNFP,
     1          FERNFR,FERNFC,BIRNMP)
      BIRTHS(INT) = FERWFC(10) + FERNFC(10)
      CALL TOTAL(RESWMC,RESWFC,RESNMC,RESNFC,POPHMC,POPHFC,6)
      TBSP = POPHMC(20) + POPHFC(20)
      DEATHS(INT) = RESPOP(INT) - TBSP + BIRTHS(INT)
      NATI(INT) = BIRTHS(INT) - DEATHS(INT)
C*
C*    READ FIXED MIGRANTS
C*
C** 1000 READ(5,910)(MIGWMC(N),N=1,19),(MIGWFC(N),N=1,19),
C**  1               (MIGNMC(N),N=1,19),(MIGNFC(N),N=1,19)
C**   THIS LOOP ZEROES OUT THE RETIREMENT MIGRANTS FOR EACH CYCLE
      DO 9311 JJ=1,19
      MIGWMC(JJ)=0.
      MIGWFC(JJ)=0.
      MIGNMC(JJ)=0.
9311  MIGNFC(JJ)=0.
      TABLE = 10
C*
C*    ADD MIGRANTS TO RESIDENT POPULATION, TOTAL POPULATION
C*
 1020 CALL TOTAL(MIGWMC,MIGWFC,MIGNMC,MIGNFC,POPHMC,POPHFC,TABLE)
      IF (TABLE.EQ.9)    GO TO 1400
      MIGR(INT) = POPHMC(20) + POPHFC(20)
C*
C*    READ SPECIAL POPULATIONS, ADD TO CALCULATED POPULATIONS
C*
1049  TABLE = 7
 1050 READ(15,910)  (SPCWM1(N),N=1,19),(SPCWF1(N),N=1,19),
     1               (SPCNM1(N),N=1,19),(SPCNF1(N),N=1,19)
      IF (TABLE.EQ.8)    GO TO 1150
      CALL TOTAL(SPCWMI,SPCWFI,SPCNMI,SPCNFI,POPHMC,POPHFC,TABLE)
      DEPPOP(NUMYR) = POPHMC(20) + POPHFC(20)
C*
C*    READ LABOR FORCE PARTICIPATION RATES,SCHOOL PARTICIPATION RATES
C*
 1100 READ(12,995)  LABWMP,LABWFP,
     1              LABNMP,LABNFP
  995 FORMAT(20F3.3)
      DO 993 I=1,5
      READ(15,994) (SCHWMP(I,J),J=1,7),(SCHWFP(I,J),J=1,7)
 993  READ(15,994) (SCHNMP(I,J),J=1,7),(SCHNFP(I,J),J=1,7)
 994  FORMAT(7F3.3,2X,7F3.3)
      IF(OUTXXX.EQ.1) GO TO 7708
      CALL PRINTT(17)
      WRITE(6,7704) YEAR
      WRITE(6,7709)
 7704 FORMAT(1H0,2X,I4,8X,'SCHOOL PARTICIPATION RATES',//)
 7709 FORMAT(4X,'AGE       WHITE     WHITE    NONWHITE   NONWHITE',/,
     *      3X,'GROUP       MALE    FEMALE      MALE      FEMALE')
      DO 7706 I=1,5
      WRITE(6,7705) (TS(J,I),J=1,3)
 7705 FORMAT(/,2X,3A4)
      DO 7706 J=1,7
      WRITE(6,7707) SAGE(J),SCHWMP(I,J),SCHWFP(I,J),SCHNMP(I,J),
```

```
     *    SCHNFP(I,J)
 7706 CONTINUE
 7707 FORMAT(2X,A8,4X,F5.3,4X,F5.3,4X,F5.3,6X,F5.3)
 7708 CONTINUE
C*
C*    READ PERCENT GROUP QUARTERS (MILITARY(MALE ONLY),
C*        COLLEGE DORMS,OTHER)
C*
      READ(15,997) GQMWMP,GQMWMP,GQCWMP,GQCWFP,GQCNMP,
     *             GQCNFP,GQOWMP,GQOWFP,GQONMP,GQONFP
  997 FORMAT((10F3.3,2X,10F3.3))
      IF(OUTXXX.EQ.1) GO TO 7751
      CALL PRINTT(18)
      WRITE(6,7710) YEAR
 7710 FORMAT(1H0,2X,I4,8X,'PERCENT GROUP QUARTERS',//)
      WRITE(6,7709)
      WRITE(6,7705) (KGQ(J,1),J=1,3)
      WRITE(6,7707)(GA(J),GQMWMP(J),GQMWFP(J),GQMNMP(J),GQMNFP(J),
     *              J=1,10)
      WRITE(6,7705) (KGQ(J,2),J=1,3)
      WRITE(6,7707)(GA(J),GQCWMP(J),GQCWFP(J),GQCNMP(J),GQCNFP(J),
     *              J=1,10)
      WRITE(6,7705) (KGQ(J,3),J=1,3)
      WRITE(6,7707)(GA(J),GQOWMP(J),GQOWFP(J),GQONMP(J),GQONFP(J),
     *              J=1,10)
 7751 CONTINUE
C*
C*    READ PERCENT HEADS OF HOUSEHOLDS
C*
      READ(14,111) HHWMP,HHWFP,HHNMP,HHNFP
111   FORMAT (10F3.3)
      IF(OUTXXX.EQ.1) GO TO 7752
      CALL PRINTT(19)
      WRITE(6,7711) YEAR
 7711 FORMAT(1H0,2X,I4,8X,'PERCENT HEADS OF HOUSEHOLDS',//)
      WRITE(6,7709)
      WRITE(6,7714)
 7714 FORMAT('     ')
      DO 7712 I=1,10
      WRITE(6,7707) GA(I),HHWMP(I),HHWFP(I),HHNMP(I),HHNFP(I)
 7712 CONTINUE
 7752 CONTINUE
C*
C*    READ AEROSPACE EMPLOYMENT,UNEMPLOYMENT RATE,
C*    JOBS PER EMPLOYEE,VACANCY RATE,PERCENT PUBLIC SCHOOLS
C*
      READ(5,996)          MIGTOT,UNMTBP,JPEMP,VR,PCTPS,KBASIC,KLFPR
  996 FORMAT(F10.0,6F10.4)
C**  THIS IS A DESPERATE ATTEMPT AT DEBUGGING
      WRITE(6,996) MIGTOT,UNMTBP,JPEMP,VR,PCTPS,KBASIC,KLFPR
      AR = 0.
      IF(KLFPR.EQ.0.) KLFPR = 1.
      IF(KBASIC.EQ.0.) KBASIC = 1.
      DO 6472 I=1,20
      LABWMP(I) = KLFPR*LABWMP(I)
      LABWFP(I) = KLFPR*LABWFP(I)
      LABNMP(I) = KLFPR*LABNMP(I)
      LABNFP(I) = KLFPR*LABNFP(I)
 6472 CONTINUE
C*
C*    CALCULATE AVAILABLE LABOR FORCE
```

```
C*
      CALL LABOR(LABWMP,TOTWMC,LFWM,LABWMC)
      CALL LABOR(LABWFP,TOTWFC,LFWF,LABWFC)
      CALL LABOR(LABNMP,TOTNMC,LFNM,LABNMC)
      CALL LABOR(LABNFP,TOTNFC,LFNF,LABNFC)
      LABTBC =(LABWMC + LABWFC)          + (LABNMC + LABNFC)
C*
C*    CALCULATE MIGRANTS TO FILL LABOR FORCE
C*
 1120 CALL LABOR(LABWMP,MIGWMP,LFWM,LABWMC)
      CALL LABOR(LABWFP,MIGWFP,LFWF,LABWFC)
      CALL LABOR(LABNMP,MIGNMP,LFNM,LABNMC)
      CALL LABOR(LABNFP,MIGNFP,LFNF,LABNFC)
      RESWBP =(RESWMC(20) + RESWFC(20)) / (RESWMC(20)+RESWFC(20) +
     1          RESNMC(20) + RESNFC(20))
      RESNBP = 1. - RESWBP
      MULTX=  1. /((LABWMC + LABWFC)*RESWBP +(LABNMC+LABNFC)*RESNBP)
C*
C*    CALCULATE SCHOOL ENROLLMENT
C*
 1121 CONTINUE
      CALL SCHOOL (SCHWMP,TOTWMC,SCHWME,SCHTWM)
      CALL SCHOOL (SCHWFP,TOTWFC,SCHWFE,SCHTWF)
      CALL SCHOOL (SCHNMP,TOTNMC,SCHNME,SCHTNM)
      CALL SCHOOL (SCHNFP,TOTNFC,SCHNFE,SCHTNF)
      SCHTBC =(SCHTWM + SCHTWF + SCHTNM + SCHTNF)
      ENROL = SCHTBC
      CEN = SCHWME(5,8)+SCHWFE(5,8)+SCHNME(5,8)+SCHNFE(5,8)
      IF(TABLE.EQ.12) GO TO 1401
      T = YEAR - 1949
      PCTSC = .86336 - .13028*ALOG(T)
      SCETBC = CEN*PCTSC
      SCE = SCETBC
C*
C*    CALCULATE RATIO OF MIGRANTS ENROLLED IN SCHOOL TO TOTAL MIGRANTS
C*
      CALL SCHOOL (SCHWMP,MIGWMP,SCHMTE,SCHWMC)
      CEN1 = SCHMTE(5,8)
      CALL SCHOOL (SCHWFP,MIGWFP,SCHMTE,SCHWFC)
      CEN2 = SCHMTE(5,8)
      CALL SCHOOL (SCHNMP,MIGNMP,SCHMTE,SCHNMC)
      CEN3 = SCHMTE(5,8)
      CALL SCHOOL (SCHNFP,MIGNFP,SCHMTE,SCHNFC)
      CEN4 = SCHMTE(5,8)
      MULTY = (SCHWMC+SCHWFC)*RESWBP + (SCHNMC+SCHNFC)*RESNBP
C*
C*    CALCULATE RATIO OF MIGRANTS ENROLLED IN COLLEGE TO TOTAL MIGRANTS
C*
      MULTW = (CEN1+CEN2)*RESWBP + (CEN3+CEN4)*RESNBP
C*    READ SPECIAL POPULATION NOT IN LABOR FORCE
      TABLE = 8
      GO TO 1050
 1150 REWIND 15
C*
C*    CALCULATE GROUP QUARTERS
C*
 1129 CONTINUE
      DO 1130 J=1,7
      CPWM(J) = SCHWMP(5,J)
      CPWF(J) = SCHWFP(5,J)
      CPNM(J) = SCHNMP(5,J)
```

```
      CPNF(J) = SCHNFP(5,J)
      CEWM(J) = SCHWME(5,J)
      CEWF(J) = SCHWFE(5,J)
      CENM(J) = SCHNME(5,J)
 1130 CENF(J) = SCHNFE(5,J)
      CALL GROUPQ(GQMWM,GQMWMP,SPCWMI,GQMNM,GQMNMP,SPCNMI)
      CALL GROUPQ(GQCDWM,GQCWMP,CEWM,GQCDWF,GQCWFP,CEWF)
      CALL GROUPQ(GQCDNM,GQCNMP,CENM,GQCDNF,GQCNFP,CENF)
      CALL GROUPQ(GQOWM,GQOWMP,TOTWMC,GQOWF,GQOWFP,TOTWFC)
      CALL GROUPQ(GQONM,GQONMP,TOTNMC,GQONF,GQONFP,TOTNFC)
      TGQ=GQCDWM(11)+GQCDWF(11)+GQCDNM(11)+GQCDNF(11)
     *    +GQOWM(11)+GQOWF(11)+GQONM(11)+GQONF(11)
     *    +GQMWM(11)+GQMNM(11)
      IF(TABLE.EQ.14) GO TO 1405
 1131 CONTINUE
C*
C*    CALCULATE HOUSEHOLD POPULATION
C*
      DO 1135 I=1,20
      TOTWMT(I) = TOTWMC(I) + SPCWMI(I)
      TOTWFT(I) = TOTWFC(I) + SPCWFI(I)
      TOTNMT(I) = TOTNMC(I) + SPCNMI(I)
      TOTNFT(I) = TOTNFC(I) + SPCNFI(I)
 1135 CONTINUE
      CALL GROUPQ(HHPWM,GQP,TOTWMT,HHPWF,GQP,TOTWFT)
      CALL GROUPQ(HHPNM,GQP,TOTNMT,HHPNF,GQP,TOTNFT)
      DO 1136 I=1,11
      HHPWM(I) = HHPWM(I) -(GQMWM(I)+GQCDWM(I)+GQOWM(I))
      HHPWF(I) = HHPWF(I) - (GQMWF(I)+GQCDWF(I)+GQOWF(I))
      HHPNM(I) = HHPNM(I) - (GQMNM(I)+GQCDNM(I)+GQONM(I))
      HHPNF(I) = HHPNF(I) - (GQMNF(I)+GQCDNF(I)+GQONF(I))
 1136 CONTINUE
      HHWMC(11) = 0.
      HHWFC(11) = 0.
      HHNMC(11) = 0.
      HHNFC(11) = 0.
C*
C*    CALCULATE HEADS OF HOUSEHOLDS
C*
      DO 1137 I=1,10
      HHWMC(I) = HHWMP(I)*HHPWM(I)
      HHWFC(I) = HHWFP(I)*HHPWF(I)
      HHNMC(I) = HHNMP(I)*HHPNM(I)
      HHNFC(I) = HHNFP(I)*HHPNF(I)
      HHWMC(11) = HHWMC(11) + HHWMC(I)
      HHWFC(11) = HHWFC(11) + HHWFC(I)
      HHNMC(11) = HHNMC(11) + HHNMC(I)
      HHNFC(11) = HHNFC(11) + HHNFC(I)
 1137 CONTINUE
      HHTBC=HHWMC(11)+HHWFC(11)+HHNMC(11)+HHNFC(11)
      DU = HHTBC/(1.-VR)
      DU2 = DU
      IF(TABLE.EQ.30) GO TO 1407
C*
C*    CALCULATE RATIO OF HOUSEHOLD HEADS TO TOTAL MIGRANTS
C*
      DO 1132 K=1,10
      NN1 = GQN1(K)
      NN2 = GQN2(K)
      DO 1132 I=NN1,NN2
      MHPWMP(I) = MIGWMP(I) - (GQOWMP(K) + GQCWMP(K)*CPWM(I))*MIGWMP(I)
```

```
      MHPWFP(I) = MIGWFP(I) - (GQOWFP(K) + GQCWFP(K)*CPWF(I))*MIGWFP(I)
      MHPNMP(I) = MIGNMP(I) - (GQONMP(K) + GQCNMP(K)*CPNM(I))*MIGNMP(I)
      MHPNFP(I) = MIGNFP(I) - (GQONFP(K) + GQCNFP(K)*CPNF(I))*MIGNFP(I)
 1132 CONTINUE
      CALL GROUPQ(HHWMC,HHWMP,MHPWMP,HHWFC,HHWFP,MHPWFP)
      CALL GROUPQ(HHNMC,HHNMP,MHPNMP,HHNFC,HHNFP,MHPNFP)
      MULTZ=(HHWMC(11)+HHWFC(11))*RESWBP+(HHNMC(11)+HHNFC(11))*RESNBP
C*
C*    SET UP FIRST APPROXIMATION OF POPULATION CHANGE
C*
      COUNT = 0.
      DO 1180 N=1,19
 1180 COUNT = SPCWMI(N) + SPCWFI(N) + SPCNMI(N) + SPCNFI(N) +
     1         TOTWMC(N) + TOTWFC(N) + TOTNMC(N) + TOTNFC(N) + COUNT
      POP = COUNT
      POP2 = POP
      LAST = POP
C*
C*    CALCULATE EMPLOYMENT
C*
      IF(OUTXXX.EQ.1) GO TO 1184
C**   CALL PRINTT(25)
      WRITE(6,6600) YEAR,MULTX,MULTY,MULTW,MULTZ,COUNT,SCHTBC,SCETBC,
     *              HHTBC,EPOP,PCTPS,PCTSC,VR,UNMTBP,BENR,BSCE,DU,JPEMP,
     *              TGQ,CEN,DUI,KLFPR,KBASIC,INTYRS,BDU
 6600 FORMAT(///,22X,6('**'),I6,' DETAILS OF CONVERGENCE OF POPULATION CH
     *ANGE  ',6('**'),//,4X,'MULTX = ',F6.4,7X,'MULTY = ',F6.4,8X,
     *'MULTW = ',F6.4,7X,'MULTZ = ',F6.4,/,4X,
     *'COUNT = ',F8.0,5X,'SCHTBC = ',F8.0,5X,
     *'SCETBC = ',F8.0,4X,'HHTBC = ',F8.0,/,4X,
     *'EPOP = ',F8.0,6X,'PCTPS = ',F6.4,8X,
     *'PCTSC = ',F6.4,7X,'VR = ',F6.4,/,4X,
     *'UNMTBP = ',F6.4,6X,'BENR = ',F8.0,7X,
     *'BSCE = ',F8.0,6X,'DU = ',F8.0,/,4X,
     *'JPEMP = ',F6.4,7X,'TGQ = ',F8.0,8X,'CEN = ',F8.0,7X,'DUI = ',
     *F8.0,/,4X,'KLFPR = ',F6.4,7X,'KBASIC = ',F6.4,7X,'INTYRS = ',F3.0,
     *9X,'BDU = ',F8.0,//,
     * 5X,'TER',5X,'LABXXX',4X,'LABTBC',4X,'MIGTBC',5X,'ENROL',
     * 7X,'TPS',6X,'SCE',8X,'DDU',7X,'DU2',6X,'TCON',5X,'POP2',
     * 7X,'LAST',6X,'DIFF',/)
 1184 CONTINUE
 1185 IF(JK.EQ.7) GO TO 27
      JK = JK + 1
      K = JK - 1
   27 DDU = DU2 - DUI
      TBEI=0.0
      THSI=0.0
      TBSI=0.0
      TCON=0.0
      TPS=0.0
C**   DEBUGGING
      NI=0
      IF(NI.LT.1) GO TO 309
      DO 30 I=1,NI
      NN=C1(I)
      GO TO (31,32,30,33,34,35),NN
C* CALCULATE BASIC INDUSTRY FORECASTS
   31 E(I)=BE(I)+B(I)*(EXP(NGR(I,JK)*T)-EXP(NGR(I,1)*(BYRED -1949.)))
     1           *1000.
      E(I) = KBASIC*E(I)
      TBEI=TBEI+E(I)
```

```
      GO TO 30
C* CALCULATE STATE COLLEGE EMPLOYMENT FORECAST
   35 E(I)=BE(I)+B(I)*(EXP(NGR(I,JK)*T)-EXP(NGR(I,1)*(BYRED -1949.)))
     *      *1000. + .22748*(SCE-BSCE)
      TBEI=TBEI+E(I)
      GO TO 30
C* CALCULATE HOUSEHOLD SERVING INDUSTRY FORECASTS
   32 E(I)=BE(I)+B(I)*(POP2-EPOP)/100.
      THSI=THSI+E(I)
      GO TO 30
C* CALCULATE CONSTRUCTION EMPLOYMENT FORECAST
   33 E(I)=7422.02+.38745*(DDU/INTYRS)+281.72*(T)
      TCON=TCON+E(I)
      GO TO 30
C* CALCULATE PUBLIC SCHOOL EMPLOYMENT FORECAST
   34 E(I)=BE(I)+B(I)*(ENROL-BENR)*PCTPS/100.
      TPS=TPS+E(I)
   30 CONTINUE
      DO 40 I=1,NI
      IF(C1(I).NE.3) GO TO 40
C* CALCULATE BUSINESS SERVING INDUSTRY FORECASTS
      E(I)=BE(I)+B(I)*(TBEI+AR+THSI+TCON+TPS-(TBBEI+BAR+TBHSI+TBCON+TBPS
     1))/100.
      TBSI=TBSI+E(I)
   40 CONTINUE
  309 CONTINUE
      TBEIX=TBEI+AR
      IF(MIGIN .NE. 1) GO TO 1187
      MIGTBC = MIGTOT
      LABXXX = MIGTBC / MULTX + LABTBC
      TER = LABXXX * JPEMP * (1. - UNMTBP)
      AR = TER
      TBEIX = AR
1187  CONTINUE
      TER=(TBEIX+THSI+TBSI+TCON+TPS)
C*
C**    DEBUG PRINTING
      WRITE(6,10021) MIGTBC,MIGTOT,LABXXX,LABTBC
10021 FORMAT(' MIGTBC = ',F9.0,' MIGTOT  = ',F9.0,
     X' LABXXX = ',F9.0,' LABTBC = ',F9.0)
 1188 LABXXX =(TER  / (1. - UNMTBP))/JPEMP
      MIGTBC = MULTX * (LABXXX - LABTBC)
      ENROL = MIGTBC*MULTY + SCHTBC
      DU2 = (MIGTBC*MULTZ/(1.-VR)) + DU
      SCE = SCETBC + MULTW*MIGTBC*PCTSC
      POP2 = POP + MIGTBC
      DIFF = POP2 - LAST
      IF(OUTXXX.EQ.1) GO TO 1189
      WRITE(6,6601) TER,LABXXX,LABTBC,MIGTBC,ENROL,TPS,SCE,
     *          DDU,DU2,TCON,POP2,LAST,DIFF
 6601 FORMAT(13(1X,F9.0))
C*
 1189 IF (DIFF .GT.-100..AND.DIFF .LT. 100.) GO TO 1190
      LAST = POP2
C**   IF(NI .EQ. 0) GO TO    1190
C**   GO TO 27
1190  CONTINUE
      DUI = DU
      TOTEMP(NUMYR) = TER
      BASEMP(NUMYR) = TBEIX
      HSEMP(NUMYR) = THSI
```

```
      BSEMP(NUMYR) = TBSI
      CONEMP(NUMYR) = TCON
      PSEMP(NUMYR) = TPS
      MIGE(INT) =(MIGTBC + .5)
      MIGT(INT) = MIGE(INT) + MIGR(INT)
C**   CALL PRINTT(18)
C*    COMPUTE FORECAST SUBTOTALS BY C2
      DO 770 NN=1,10
770   DWFD(NN)=0.0
      DWFD(4)=DWFD(4)+AR
      IF(NI.LT.1) GO TO 7761
      DO 776 I=1,NI
      NN=C2(I)
776   DWFD(NN)=DWFD(NN)+E(I)
 7761 CONTINUE
      K=K+1
      TOTAL3=0.0
      NUMC2 = 0
      ESIC = 0.0
      DO 778 NN=1,10
      DWFDS(K,NN)=DWFD(NN)
778   TOTAL3=TOTAL3+DWFD(NN)
      TOTAL4(K)=TOTAL3
C**   WRITE(6,10010)YEAR
10010 FORMAT(1HO,34X,'EMPLOYMENT  FORECAST  BY  MAJOR INDUSTRY',
     1I6/1X,132('*')/)
C**   WRITE(6,10020) BYRED,LYREMP
10020 FORMAT(' ',80X,'PERCENT',8X,'ANNUAL  RATE  OF  GROWTH  FROM'/
     115X,'MAJOR  INDUSTRY',20X,'SIC',8X,'EMPLOYMENT',6X,'DISTRIBUTION',
     210X,I4,13X,I4/1X,132('*'),/)
C*
C*    COMBINE EMPLOYMENT CATEGORIES WHOSE SIC CODES ARE EQUAL
C*
      IF(NI.LT.1) GO TO 8741
      DO 874 I=1,NI
      NN=C2(I)
      IF(I.EQ.5) GO TO 877
 875  IF(SIC(I).EQ.SIC(I+1)) GO TO 873
      ESIC = ESIC + E(I)
      PCTD = (ESIC/TOTAL3)*100.
      ARGB = (ALOG(ESIC/BPERM(I)))/(YEAR-BYRED)*100.
      ARG5 = (ALOG(ESIC/ELAST(I)))/(YEAR-LYREMP)*100.
      WRITE(6,10050) IN1(I),IN2(I),IN3(I),IN4(I),IN5(I),
     *                SIC(I),ESIC,PCTD,ARGB,ARG5
10050 FORMAT(18X,4A6,A4,3X,A3,9X,F9.0,3(9X,F7.3,' '))
      NUMC2 = NUMC2 + 1
      ESIC = 0.0
      IF(NN.EQ.C2(I+1)) GO TO 874
      IF(NUMC2.EQ.1) WRITE(6,7714)
      IF(NUMC2.EQ.1) GO TO 876
      PCTD = DWFD(NN)/TOTAL3*100.
      ARGB=(ALOG(DWFD(NN  )/DWBDP(NN  )))/(YEAR-BYRED)*100.
      ARG5=(ALOG(DWFD(NN  )/DWBD(NN  )))/(YEAR-LYREMP)*100.
      WRITE(6,10030)DWFD(NN  ),PCTD,ARGB,ARG5
10030 FORMAT(30X,'SUB TOTAL',22X,F9.0,3(9X,F7.3,' ')//)
      GO TO 876
877   PCTD=(AR/TOTAL3)*100
      ARGB=(ALOG(AR/PBAR))/(YEAR-BYRED)*100.
      ARG5=(ALOG(AR/BAR))/(YEAR-LYREMP)*100.
      WRITE(6,10060) AR,PCTD,ARGB,ARG5
10060 FORMAT(18X,'AEROSPACE',23X,'19',9X,F9.0,3(9X,F7.3,' '))
```

```
      GO TO 875
  876 NUMC2 = 0
      GO TO 874
  873 ESIC = ESIC + E(I)
  874 CONTINUE
 8741 CONTINUE
      PCTD=100.0
      ARG8=(ALOG(TOTAL3/TOTAL1))/(YEAR-BYRED)*100.
      ARG5=(ALOG(TOTAL3/TOTAL2))/(YEAR-LYREMP)*100.
      WRITE(6,10070)TOTAL3,PCTD,ARG8,ARG5
10070 FORMAT(10X,'TOTAL',46X,F9.0,3(9X,F7.3,' '))
      IF(NI.LT.1) GO TO 801
      DO 80 I=1,NI
      IF(SIC(I).EQ.SIC(I+1)) GO TO 79
      ELAST(I) = ESIC + E(I)
      ESIC = 0.0
      GO TO 80
   79 ESIC = ESIC + E(I)
   80 CONTINUE
  801 CONTINUE
      DO 878 NN=1,10
  878 DWBD(NN)=DWFD(NN)
      TOTAL2=TOTAL3
C*
C**   IF(PYRMDS.EQ.0) GO TO 1195
C*    PRINT BAR GRAPH FOR EMPLOYMENT BY MAJOR INDUSTRY GROUPS
C*
C**   CALL PRINTT(11)
C**   WRITE(6,6667)
C**   WRITE(6,6667)
C**   WRITE(6,6667)
C**   WRITE(6,6667)
C**   WRITE(6,6667)
C**6667  FORMAT(1H0)
C**   DO 998 NN=1,10
C**   ID=(DWFD(NN)/3000.)+1.5
C**   IF(ID.GT.100) ID=100
C**   IF(ID.LT.1) ID=1
C**   IFMT(3)=TZ(ID+1)
C**   IF(ID.EQ.1) IX=2
C**   IX=ID
C**   JFMT(4)=TZ(IX-1)
C**   WRITE(6,IFMT)
C**   L=NN*3-2
C**   MM=NN*3
C**   IF (ID.NE.1) GO TO 6665
C**   WRITE (6,6666)(LIT(I),I=L,MM)
C**6666  FORMAT(2X,3A8,1H*)
C**   GO TO 998
C**6665WRITE(6,JFMT)(LIT(I),I=L,MM)
C**998WRITE(6,IFMT)
C**   WRITE(6,11020)
C**11020 FORMAT(26X,101('*'))
C**   WRITE(6,11030)
C**11030 FORMAT(17X,11(9X,'*'))
C**   WRITE(6,11040)
C**11040 FORMAT(26X,'0',9X,'3',9X,'6',9X,'9',8X,'12',8X,'15',8X,'18',8X,(
C**  1'21',8X,'24',8X,'27',8X,'30')
C**   WRITE(6,11050)
1105( FORMAT(1H0,50X,'EMPLOYMENT  IN  TENS  OF  THOUSANDS')
C*
```

```
C*    PRINT EMPLOYMENT FORECAST
C*
1195  IF(OUTXXX.LT.2) GO TO 1200
      CALL PRINTT(18)
      WRITE(6,9606) YEAR
 9606 FORMAT(' ',29X,'EMPLOYMENT FORECASTS'/36X,'FOR',1X,I4//21X'BASIC I
     1NDUSTRIES='//31X,'SIC',4X,'EMPLOYMENT')
      WRITE(6,9607)AR
 9607 FORMAT(/2X,'OTHER EMPLOYMENT',14X,'XX',5X,F10.0)
      IF(NI.EQ.0) GO TO 119
      DO 50 I=1,NI
      IF(NI .EQ. 0) GO TO    50
      IF((C1(I).NE.1).AND.(C1(I).NE.6)) GO TO 50
      WRITE(6,9608)IN1(I),IN2(I),IN3(I),IN4(I),IN5(I),SIC(I),E(I)
 9608 FORMAT(2X,4A6,A4,1X,A3,5X,F10.0)
   50 CONTINUE
      WRITE(6,9609)TBEIX
 9609 FORMAT(29X,'TOTAL',7X,F8.0)
      WRITE(6,9610)
 9610 FORMAT(//,21X,'HOUSEHOLD SERVING INDUSTRIES=')
      DO 60 I=1,NI
      IF(C1(I).NE.2) GO TO 60
      WRITE(6,9608)IN1(I),IN2(I),IN3(I),IN4(I),IN5(I),SIC(I),E(I)
   60 CONTINUE
      WRITE(6,9609)THSI
      WRITE(6,9611)
 9611 FORMAT(//,21X,'BUSINESS-SERVING INDUSTRIES=')
      DO 70 I=1,NI
      IF(NI .EQ. 0) GO TO    70
      IF(C1(I).NE.3)GO TO 70
      WRITE(6,9608)IN1(I),IN2(I),IN3(I),IN4(I),IN5(I),SIC(I),E(I)
   70 CONTINUE
      WRITE(6,9609)TBSI
      WRITE(6,114)
  114 FORMAT(//,21X,'CONSTRUCTION=')
      DO 117 I=1,NI
      IF(NI .EQ. 0) GO TO    117
      IF(C1(I).NE.4)GO TO 117
      WRITE(6,9608)IN1(I),IN2(I),IN3(I),IN4(I),IN5(I),SIC(I),E(I)
  117 CONTINUE
      WRITE(6,115)
  115 FORMAT(//,21X,'PUBLIC SCHOOLS=')
      DO 116 I=1,NI
      IF(NI .EQ. 0) GO TO    116
      IF(C1(I).NE.5)GO TO 116
      WRITE(6,9608)IN1(I),IN2(I),IN3(I),IN4(I),IN5(I),SIC(I),E(I)
  116 CONTINUE
119   CONTINUE
      WRITE(6,9613) TER
 9613 FORMAT(/23X,'GRAND TOTAL',F15.0)
      WRITE(6,112) POP2,DU2,ENROL
  112 FORMAT(/30X,'POPULATION          = ',F9.0,/,
     *30X,'DWELLING UNITS    = ',
     1F9.0/30X,'SCHOOL ENROLLMENT = ',F9.0)
      WRITE (6,973) YEAR,TER,   YEAR,UNMTBP,YEAR,LABXXX,YEAR,
     1              LABTBC
 973  FORMAT (1H0,I4,' EMPLOYMENT = ',F23.0,/, I5,' UNEMPLOYMENT RATE ='
     1,       F16.3          /' ',I4,' LABOR FORCE = ',      F22.0,/, I5,
     2        ' LABOR FORCE (AVAILABLE) = ',F10.0,//)
C*
C*    ALLOCATE MIGRANTS
```

```
C*
 1200 DO 1300 N=1,19
      MIGWMC(N) = MIGWMP(N) * MIGTBC    * RESWBP
      MIGWFC(N) = MIGWFP(N) * MIGTBC    * RESWBP
      MIGNMC(N) = MIGNMP(N) * MIGTBC    * RESNBP
      MIGNFC(N) = MIGNFP(N) * MIGTBC    * RESNBP
 1300 CONTINUE
      TABLE = 9
      GO TO 1020
 1400 TABLE = 12
      GO TO 1121
C*
C*    PRINT SCHOOL ENROLLMENT, GROUP QUARTERS,
C*    HOUSEHOLD POPULATION, HOUSEHOLD HEADS
C*
 1401 DO 1402 I=1,6
      DO 1402 J=1,8
      SCHMTE(I,J) = SCHWME(I,J) + SCHNME(I,J)
      SCHFTE(I,J) = SCHWFE(I,J) + SCHNFE(I,J)
 1402 CONTINUE
      IF(OUTXXX.EQ.1) GO TO 1404
      IF(RACES.EQ.1) GO TO 1403
      CALL PRINTS(SCHWME,SCHWFE,1)
      CALL PRINTS(SCHNME,SCHNFE,2)
 1403 CALL PRINTS(SCHMTE,SCHFTE,0)
 1404 CONTINUE
      SCHENR(NUMYR) = SCHTBC
      SCHNUR(NUMYR) = SCHMTE(1,8) + SCHFTE(1,8)
      SCHKIN(NUMYR) = SCHMTE(2,8) + SCHFTE(2,8)
      SCHEL(NUMYR) = SCHMTE(3,8) + SCHFTE(3,8)
      SCHHS(NUMYR) = SCHMTE(4,8) + SCHFTE(4,8)
      SCHCOL(NUMYR) = SCHMTE(5,8) + SCHFTE(5,8)
      TABLE = 14
      GO TO 1129
 1405 CONTINUE
      DO 1406 I=1,11
      GQWMT(I) = GQMWM(I) + GQCDWM(I) + GQOWM(I)
      GQWFT(I) = GQCDWF(I) + GQOWF(I)
      GQNMT(I) = GQMNM(I) + GQCDNM(I) + GQONM(I)
      GQNFT(I) = GQCDNF(I) + GQONF(I)
      GQMMT(I) = GQMWM(I) + GQMNM(I)
      GQMFT(I) = GQMWF(I) + GQMNF(I)
      GQCDMT(I) = GQCDWM(I) + GQCDNM(I)
      GQCDFT(I) = GQCDWF(I) + GQCDNF(I)
      GQOMT(I) = GQOWM(I) + GQONM(I)
      GQOFT(I) = GQOWF(I) + GQONF(I)
      GQMT(I) = GQWMT(I) + GQNMT(I)
      GQFT(I) = GQWFT(I) + GQNFT(I)
 1406 CONTINUE
      IF(OUTXXX.EQ.1) GO TO 1411
      CALL PRINTG(GQMWM,GQMWF,1,1)
      CALL PRINTG(GQMNM,GQMNF,1,2)
      CALL PRINTG(GQMMT,GQMFT,1,0)
      CALL PRINTG(GQCDWM,GQCDWF,2,1)
      CALL PRINTG(GQCDNM,GQCDNF,2,2)
      CALL PRINTG(GQCDMT,GQCDFT,2,0)
      CALL PRINTG(GQOWM,GQOWF,3,1)
      CALL PRINTG(GQONM,GQONF,3,2)
      CALL PRINTG(GQOMT,GQOFT,3,0)
      CALL PRINTG(GQWMT,GQWFT,4,1)
      CALL PRINTG(GQNMT,GQNFT,4,2)
```

```
      CALL PRINTG(GQMT,GQFT,4,0)
 1411 CONTINUE
      GQMB(NUMYR) = GQMWM(11) + GQMNM(11)
      GQCD(NUMYR) = GQCDWM(11) + GQCDWF(11) + GQCDNM(11) + GQCDNF(11)
      GQO(NUMYR) = GQOWM(11) + GQOWF(11) + GQONM(11) + GQONF(11)
      TOTGQ(NUMYR) = GQMB(NUMYR) + GQCD(NUMYR) + GQO(NUMYR)
      TABLE = 30
      GO TO 1131
 1407 CONTINUE
      DO 1408 I=1,11
      HHMTC(I) = HHWMC(I) + HHNMC(I)
      HHFTC(I) = HHWFC(I) + HHNFC(I)
      HHPMTC(I) = HHPWM(I) + HHPNM(I)
      HHPFTC(I) = HHPWF(I) + HHPNF(I)
 1408 CONTINUE
      IF(OUTXXX.EQ.1) GO TO 1409
      CALL PRINTG(HHPWM,HHPWF,6,1)
      CALL PRINTG(HHPNM,HHPNF,6,2)
      CALL PRINTG(HHPMTC,HHPFTC,6,0)
      CALL PRINTG(HHWMC,HHWFC,5,1)
      CALL PRINTG(HHNMC,HHNFC,5,2)
      CALL PRINTG(HHMTC,HHFTC,5,0)
 1409 CONTINUE
      HHPOP(NUMYR) = HHPMTC(11) + HHPFTC(11)
      HHH(NUMYR) = HHMTC(11) + HHFTC(11)
      CALL TOTAL(SPCWMI,SPCWFI,SPCNMI,SPCNFI,POPHMC,POPHFC,8)
      MISPOP(NUMYR) = POPHMC(20) + POPHFC(20)
 1410 CALL TOTAL(TOTWMC,TOTWFC,TOTNMC,TOTNFC,POPHMC,POPHFC,0)
      TOTPOP(NUMYR) = POPHMC(20) + POPHFC(20)
      TOTCH(INT)=TOTPOP(NUMYR)-TOTPOP(NUMYR-1)
      MILREL(INT)=(MISPOP(NUMYR)+DEPPOP(NUMYR))
     *              -(MISPOP(NUMYR-1) + DEPPOP(NUMYR-1))
C*
C*    CALL ROUTINE TO CALCULATE AND PRINT CHANGES
C*
 1500 CALL CHANGE(TOTWMI,TOTWFI,TOTWMC,TOTWFC,0,1)
      CALL CHANGE(TOTNMI,TOTNFI,TOTNMC,TOTNFC,0,2)
      CALL CHANGE(TOTBMI,TOTBFI,TOTBMC,TOTBFC,0,0)
      CALL PYRMID(TOTWMC,TOTWFC,21,0,1,YEAR)
      CALL PYRMID(TOTWMI,TOTWFI,22,0,1,LASTYR)
      CALL PYRMID(TOTNMC,TOTNFC,21,0,2,YEAR)
      CALL PYRMID(TOTNMI,TOTNFI,22,0,2,LASTYR)
      CALL PYRMID(TOTBMC,TOTBFC,21,0,0,YEAR)
      CALL PYRMID(TOTBMI,TOTBFI,22,0,0,LASTYR)
C*
C*    WRAPUP CYCLE
C*
      DO 8900 N=1,20
      RESWMI(N) = RESWMC(N)
      RESWFI(N) = RESWFC(N)
      RESNMI(N) = RESNMC(N)
      RESNFI(N) = RESNFC(N)
      TOTWMI(N) = TOTWMC(N)
      TOTWFI(N) = TOTWFC(N)
      TOTNMI(N) = TOTNMC(N)
      TOTNFI(N) = TOTNFC(N)
      TOTBMI(N) = TOTBMC(N)
      TOTBFI(N) = TOTBFC(N)
 8900 CONTINUE
      OWTOT(K)=TOTBMI(20)+TOTBFI(20)
      DO 1420 I=1,19
```

```
      TPLSWM(I) = TOTWMC(I) - SPCWMI(I)
      TPLSWF(I) = TOTWFC(I) - SPCWFI(I)
      TPLSNM(I) = TOTNMC(I) - SPCNMI(I)
      TPLSNF(I) = TOTNFC(I) - SPCNFI(I)
 1420  CONTINUE
      CALL LABOR(LABWMP,TPLSWM,LFWM,LABWMC)
      CALL LABOR(LABWFP,TPLSWF,LFWF,LABWFC)
      CALL LABOR(LABNMP,TPLSNM,LFNM,LABNMC)
      CALL LABOR(LABNFP,TPLSNF,LFNF,LABNFC)
      LFTM(20) = 0.0
      LFTF(20) = 0.0
      DO 1450 I=1,19
      LFTM(I) = LFWM(I) + LFNM(I)
      LFTF(I) = LFWF(I) + LFNF(I)
      LFTM(20) = LFTM(20) + LFTM(I)
      LFTF(20) = LFTF(20) + LFTF(I)
 1450 CONTINUE
      IF(OUTXXX.EQ.1) GO TO 1460
      CALL PRINTR(LFWM,LFWF,1,11,1,0)
      CALL PRINTR(LFNM,LFNF,1,11,2,0)
      CALL PRINTR(LFTM,LFTF,1,11,0,0)
 1460 LABFOR(NUMYR) = LFTM(20) + LFTF(20)
C*
C*    READ CYCLE CARD
C*
      IF (YEAR .LT. ENDYR) GO TO 400
C*
C*    END OF      PRINT CHANGES FROM BASE YEAR TO FINAL YEAR
C*
 1900 LASTYR = BASEYR
      CALL CHANGE(TOTWMO,TOTWFO,TOTWMI,TOTWFI,0,1)
      CALL CHANGE(TOTNMO,TOTNFO,TOTNMI,TOTNFI,0,2)
      CALL CHANGE(TOTBMO,TOTBFO,TOTBMI,TOTBFI,0,0)
      CALL PYRMID(TOTWMI,TOTWFI,21,0,1,YEAR)
      CALL PYRMID(TOTWMO,TOTWFO,22,0,1,LASTYR)
      CALL PYRMID(TOTNMI,TOTNFI,21,0,2,YEAR)
      CALL PYRMID(TOTNMO,TOTNFO,22,0,2,LASTYR)
      CALL PYRMID(TOTBMI,TOTBFI,21,0,0,YEAR)
      CALL PYRMID(TOTBMO,TOTBFO,22,0,0,LASTYR)
C*
 7532 CONTINUE
      CALL TOTAL(RESWMI,RESWFI,RESNMI,RESNFI,POPHMC,POPHFC,1)
      RESPOP(INT+1) = POPHMC(20) + POPHFC(20)
      RESCH(INT) = RESPOP(NUMYR) - RESPOP(NUMYR-1)
      HHPOP(1) = TOTPOP(1) - TOTGQ(1)
      IF(NUMYR.GT.11) NUMYR = 11
      DO 9900 I=1,NUMYR
      TOTPOP(I) = TOTPOP(I)/100*100
      RESPOP(I) = RESPOP(I)/100*100
      MISPOP(I) = MISPOP(I)/100*100
      DEPPOP(I) = DEPPOP(I)/100*100
      TOTEMP(I) = TOTEMP(I)/100*100
      BASEMP(I) = BASEMP(I)/100*100
      HSEMP(I) = HSEMP(I)/100*100
      BSEMP(I) = BSEMP(I)/100*100
      CONEMP(I) = CONEMP(I)/100*100
      PSEMP(I) = PSEMP(I)/100*100
      LABFOR(I) = LABFOR(I)/100*100
      SCHENR(I) = SCHENR(I)/100*100
      SCHNUR(I) = SCHNUR(I)/100*100
      SCHKIN(I) = SCHKIN(I)/100*100
```

```
      SCHEL(I) = SCHEL(I)/100*100
      SCHHS(I) = SCHHS(I)/100*100
      SCHCOL(I) = SCHCOL(I)/100*100
      HHPOP(I) = HHPOP(I)/100*100
      HHH(I) = HHH(I)/100*100
      TOTGQ(I) = TOTGQ(I)/100*100
      GQMB(I) = GQMB(I)/100*100
      GQCD(I) = GQCD(I)/100*100
      GQO(I) = GQO(I)/100*100
 9900 CONTINUE
      CALL PRINTT(12)
      WRITE(6,9801)
 9801 FORMAT(1HO,42X,'COMPONENTS OF POPULATION GROWTH',//,
     *  52X,'S U M M A R Y',///)
      NYR = NUMYR - 1
      WRITE(6,9802) (SUMMYR(I),SUMMYR(I+1),I=1,NYR)
 9802 FORMAT(21X,10(I4,'-',I4,2X))
      WRITE(6,9803) (TOTCH(I),I=1,NYR)
 9803 FORMAT(/,1X,'TOTAL CHANGE',5X,10I11)
      WRITE(6,9804) (RESCH(I),I=1,NYR)
 9804 FORMAT(/,2X,'RESIDENT',8X,10I11)
      WRITE(6,9805) (NATI(I),I=1,NYR)
 9805 FORMAT(/,3X,'NATURAL INCREASE',I10,9I11)
      WRITE(6,9806) (BIRTHS(I),I=1,NYR)
 9806 FORMAT(4X,'BIRTHS',8X,10I11)
      WRITE(6,9807) (DEATHS(I),I=1,NYR)
 9807 FORMAT(4X,'DEATHS',8X,10I11)
      WRITE(6,9808) (MIGT(I),I=1,NYR)
 9808 FORMAT(/,3X,'MIGRATION',6X,10I11)
      WRITE(6,9809) (MIGE(I),I=1,NYR)
 9809 FORMAT(4X,'EMPLOYMENT',4X,10I11)
      WRITE(6,9810) (MIGR(I),I=1,NYR)
 9810 FORMAT(4X,'RETIREMENT',4X,10I11)
      WRITE(6,9811) (MILREL(I),I=1,NYR)
 9811 FORMAT(/,2X,'MILITARY RELATED',10I11)
      CALL PRINTT(12)
      WRITE(6,9901)
 9901  FORMAT(1HO,43X,'F O R E C A S T   S U M M A R Y',///)
      WRITE(6,9902)(SUMMYR(I),I=1,NUMYR)
 9902  FORMAT(30X,11(1X,I8))
      WRITE(6,9903)(TOTPOP(I),I=1,NUMYR)
 9903 FORMAT(//,1X,'TOTAL POPULATION',13X,11(1X,I8))
      WRITE(6,9904)(RESPOP(I),I=1,NUMYR)
 9904 FORMAT(3X,'RESIDENT POP.',14X,11(1X,I8))
      WRITE(6,9905)(MISPOP(I),I=1,NUMYR)
 9905 FORMAT(3X,'MILITARY IN-SERVICE POP.',3X,11(1X,I8))
      WRITE(6,9906)(DEPPOP(I),I=1,NUMYR)
 9906 FORMAT(3X,'MILITARY DEPENDENT POP.',4X,11(1X,I8))
      IF(BASEYR.NE.BYRED) GO TO 9930
      WRITE(6,9907)(TOTEMP(I),I=1,NUMYR)
 9907 FORMAT(//,1X,'TOTAL EMPLOYMENT',13X,11(1X,I8))
      GO TO 9932
 9930 WRITE(6,9931) (TOTEMP(I),I=1,NUMYR)
 9931 FORMAT(//,1X,'TOTAL EMPLOYMENT *',11X,11(1X,I8))
 9932 CONTINUE
      WRITE(6,9908)(BASEMP(I),I=1,NUMYR)
 9908 FORMAT(3X,'BASIC EMP.',17X,11(1X,I8))
      WRITE(6,9909)(HSEMP(I),I=1,NUMYR)
 9909 FORMAT(3X,'HOUSEHOLD-SERVING EMP.',5X,11(1X,I8))
      WRITE(6,9910)(BSEMP(I),I=1,NUMYR)
 9910 FORMAT(3X,'BUSINESS-SERVING EMP.',6X,11(1X,I8))
```

```
      WRITE(6,9926) (CONEMP(I),I=1,NUMYR)
 9926 FORMAT(3X,'CONSTRUCTION EMP.',10X,11(1X,I8))
      WRITE(6,9927) (PSEMP(I),I=1,NUMYR)
 9927 FORMAT(3X,'PUBLIC SCHOOL EMP.',9X,11(1X,I8))
      WRITE(6,9911)(LABFOR(I),I=1,NUMYR)
 9911 FORMAT(//,1X,'LABOR FORCE',18X,11(1X,I8))
      WRITE(6,9912)(SCHENR(I),I=1,NUMYR)
 9912 FORMAT(//,1X,'TOTAL SCHOOL ENROLLMENT',6X,11(1X,I8))
      WRITE(6,9913)(SCHNUR(I),I=1,NUMYR)
 9913 FORMAT(3X,'NURSERY',20X,11(1X,I8))
      WRITE(6,9914)(SCHKIN(I),I=1,NUMYR)
 9914 FORMAT(3X,'KINDERGARTEN',15X,11(1X,I8))
      WRITE(6,9915)(SCHEL(I),I=1,NUMYR)
 9915 FORMAT(3X,'ELEMENTARY',17X,11(1X,I8))
      WRITE(6,9916)(SCHHS(I),I=1,NUMYR)
 9916 FORMAT(3X,'HIGH SCHOOL',16X,11(1X,I8))
      WRITE(6,9917)(SCHCOL(I),I=1,NUMYR)
 9917 FORMAT(3X,'COLLEGE',20X,11(1X,I8))
      WRITE(6,9925) (HHPOP(I),I=1,NUMYR)
 9925 FORMAT(//,1X,'HOUSEHOLD POPULATION',9X,11(1X,I8))
      WRITE(6,9918)(HHH(I),I=1,NUMYR)
 9918 FORMAT(/,1X,'HOUSEHOLD HEADS',14X,11(1X,I8))
      WRITE(6,9919)(TOTGQ(I),I=1,NUMYR)
 9919 FORMAT(//,1X,'TOTAL GROUP QUARTERS',9X,11(1X,I8))
      WRITE(6,9920)(GQMB(I),I=1,NUMYR)
 9920 FORMAT(3X,'MILITARY BARRACKS',10X,11(1X,I8))
      WRITE(6,9921)(GQCD(I),I=1,NUMYR)
 9921 FORMAT(3X,'COLLEGE DORMITORIES',8X,11(1X,I8))
      WRITE(6,9922)(GQO(I),I=1,NUMYR)
 9922 FORMAT(3X,'OTHER GROUP QUARTERS',7X,11(1X,I8))
      IF(BASEYR.NE.BYRED) WRITE(6,9940) BYRED
 9940 FORMAT(/////,5X,'* EMPLOYMENT BASE YEAR = ',I4)
      STOP
      END
      SUBROUTINE TITLEP
C*    THIS SUBROUTINE PRODUCES THE PROGRAM TITLE AND CREDITS
C*
      WRITE (6,1)
      WRITE (6,3)
      WRITE (6,4)
      WRITE (6,41)
      WRITE (6,5)
      WRITE (6,6)
      WRITE (6,7)
      WRITE (6,71)
      WRITE (6,8)
      WRITE (6,81)
      WRITE (6,9)
      WRITE (6,91)
1     FORMAT ('1     ')
3     FORMAT (/,46X,'   AAA      BBBBBB        AAA          GGG     ',/,46X,
     A               ' AA   AA   BBB  BB     AA   AA     GG  GG    ',/,46X,
     A               'AAA   AAA  BBB  BBB   AAA   AAA   GGG  GGG   ',/,46X,
     A               'AAA   AAA  BBB  BBB   AAA   AAA   GGG         ',/,46X,
     A               'AAA   AAA  BBBBBBB    AAA   AAA   GGG         ',/,46X,
     A               'AAAAAAAAA  BBB  BBB   AAAAAAAAA   GGG   GGGGG ',/,46X,
     A               'AAA   AAA  BBB  BBB   AAA   AAA   GGG    GGG  ',/,46X,
     A               'AAA   AAA  BBB  BBB   AAA   AAA   GGG    GGG  ',/,46X,
     A               'AAA   AAA  BBB  BBB   AAA   AAA    GGG  GGG   ',/,46X,
     A               'AAA   AAA  BBBBBBB    AAA   AAA     GGGGGGG  ',////)
4     FORMAT (/,6X,'PPPPP OOOOO PPPPP U   U L      AAAAA TTTTT  I  OOOOO
```

```
      A N    N',7X,'PPPPP RRRRR OOOOO  JJJJJ EEEEE CCCCC TTTTT  OOOOO RRR
     8RR',/,6X,
      C              'P   P O   O P   P U   U L     A   A   T    I O   O
      D N N  N',7X,'P   P R   R O   O    J   E     C   C   T    O   O R
      E R',/,6X,
      F              'P   P O   O P   P U   U L     A   A   T    I O   O
      G N  N N',7X,'P   P R   R O   O    J   E     C       T    O   O R
      H R',/,6X,
      I              'P   P O   O P   P U   U L     A   A   T    I O   O
      J N    N',7X,'P   P R   R O   O    J   E     C       T    O   O R
      K R',/,6X,
      L              'PPPPP O   O PPPPP U   U L     AAAAA   T    I O   O
      M N    N',7X,'PPPPP RRRRR O   O    J   EEEE  C       T    O   O RRR
     NRR')
41    FORMAT (6X,  'P     O   O P     U   U L     A   A   T    I O   O
      P N    N',7X,'P     R R   O   O    J   E     C       T    O   O R R
      Q  ',/,6X,
      R              'P     O   O P     U   U L   L A   A   T    I O   O
      S N    N',7X,'P     R  R  O   O J  J  E     C   C   T    O   O R
     TR ',/,6X,
      U              'P     OOOOO P     UUUUU LLLLL A   A   T    I OOOOO
      V N    N',7X,'P     R   R OOOOO JJJJJ  EEEEE CCCCC   T    OOOOO R
      W R')
5     FORMAT (////,57X,'AAAAA                ',/,57X,
     X                 'AA AA                ',/,57X,
     1                 'AA AA NN  NN         ',/,57X,
     2                 'AA AA NNN NN         ',/,57X,
     3                 'AAAAA NN NNN DDDD    ',/,57X,
     4                 'AA AA NN  NN DD DD   ',/,57X,
     5                 'AA AA NN  NN DD DD   ',/,57X,
     6                 'AA AA NN  NN DD DD   ',/,57X,
     7                 '      NN  NN DD DD   ',/,57X,
     8                 '      NN  NN DD DD   ',/,57X,
     9                 '          DD DD   ',/,57X,
     A                 '          DDDD    ')
6     FORMAT (/////,19X,'L     AAAAA BBBB  OOOOO RRRR ',11X,'EEEEE  SSS
     A TTTTT I MM   MM AAAAA  TTTTT OOOOO RRRRR',/,19X,
     B                 'L     A   A B   B O   O R   R',11X,'E   E S   S
     C   T   I M M M M A   A    T   O   O R   R',/,19X,
     D                 'L     A   A B   B O   O R   R',11X,'E     S
     E   T   I M  M  M A   A    T   O   O R   R',/,19X,
     F                 'L     A   A BBBB  O   O R   R',11X,'E      SSS
     G   T   I M     M A   A    T   O   O R   R',/,19X,
     H                 'L     AAAAA B   B O   O RRRR ',11X,'EEEE      S
     I   T   I M     M AAAAA    T   O   O RRRRR',/,19X,
     J                 'L     A   A B   B O   O R R  ',11X,'E         S
     K   T   I M     M A   A    T   O   O R R  ',/,19X,
     L                 'L   L A   A B   B O   O R  R ',11X,'E   E S   S
     M   T   I M     M A   A    T   O   O R  R ',/,19X,
     N                 'LLLLL A   A BBBB  OOOOO R   R',11X,'EEEEE  SSS
     O   T   I M     M A   A    T   OOOOO R   R')
7     FORMAT ('1',////,42X,'   AAA     PPPPPP    PPPPPP    LLL      EEEE
     AEEE',/,42X,
     B                  ' AA   AA   PPP  PP   PPP  PP   LLL      EEEE
     CEEE',/,42X,
     D                  'AAA   AAA  PPP  PPP  PPP  PPP  LLL      EEE
     E   ',/,42X,
     F                  'AAA   AAA  PPP  PPP  PPP  PPP  LLL      EEE
     G   ',/,42X,
     H                  'AAA   AAA  PPP  PPP  PPP  PPP  LLL      EEEE
     1EE ')
```

```
 71    FORMAT(42X,           'AAAAAAAAA  PPPPPPP   PPPPPPP   LLL       EEEE
      KEE ',/,42X,
      L                     'AAA   AAA  PPP       PPP       LLL       EE
      K   ',/,42X,
      N                     'AAA   AAA  PPP       PPP       LLL       EEE
      M   ',/,42X,
      P                     'AAA   AAA  PPP       PPP       LLL LLL   EEEE
      OEEE',/,42X,
      R                     'AAA   AAA  PPP       PPP       LLLLLLL   EEEE
      QEEE',//)
  8    FORMAT (50X,  '              OO              ',/,
      A          50X,'             OO .. .          ',/,
      B          50X,'     ....... OO..    ....     ',/,
      C          50X,'  ..       ..OO          ..   ',/,
      D          50X,' .                      .  ',/,
      E          50X,'.                      .  ',/,
      F         49X,'.                      .  ',/,
      G        48X,'.                      .  ',/,
      H        48X,'.                      .  ',/,
      I        48X,'.                      .  ',/,
      J        48X,'.                      .  ',/,
      K         49X,'.                      .  ',/,
      L          50X,'.                    .  ',/,
      M          50X,' .                  .  ',/,
      N          50X,'  .                .  ',/,
      O          50X,'   .              .  ')
 81    FORMAT (  50X,'    .            .  ',/,
      Q          50X,'     .          .  ',/,
      R          50X,'      .   ...   .  ',/,
      S          50X,'       ..  ... ......  ',/,
      T          50X,'         ...           ')
  9    FORMAT (/////,14X,'LL      AAAAA  NN NN  KK  KK  FFFFF   OOOOO   RRRR
      *R  DDDD ',
      A    14X,' SSS    HH HH   YY  YY  EEEEE  RRRRR',/,14X,
      B             'LL      AA AA  NNNNN  KK KK   FFFFF  OO OO  RR RR  DD D
      CD',14X,'SS SS   HH HH   YY  YY  EE EE  RR RR',/,14X,
      D             'LL      AA AA  NN NN  KKKK    FF     OO OO  RR RR  DD D
      ED',14X,'SS      HH HH   YY  YY  EE      RR RR',/,14X,
      D             'LL      AA AA  NN NN  KKKK    FF     OO OO  RR RR  DD D
      ED',14X,'SS      HH HH   YY  YY  EE      RR RR',/,14X,
      F             'LL      AA AA  NN NN  KKK     FFFF   OO OO  RR RR  DD D
      GD',14X,' SSS    HH HH    YYYY   EEEE   RR RR',/,14X,
      H             'LL      AAAAA  NN NN  KKKK    FFFF   OO OO  RRRRR  DD D
      ID',14X,'   SS   HHHHH     YY    EEEE   RRRRR',/,14X,
      J             'LL LL   AA AA  NN NN  KK KK   FF     OO OO  RRR    DD D
      KD',14X,'   SS   HH HH     YY    EE     RRR  ')
 91    FORMAT(14X,'LL LL   AA AA  NN NN  KK KK   FF     OO OO  RRR    DD D
      KD',14X,'   SS   HH HH     YY    EE     RRR  ',/,14X,
      L             'LLLLL   AA AA  NN NN  KK  KK  FF     OO OO  RR R   DD D
      MD',14X,'SS SS   HH HH     YY    EE EE  RR R ',/,14X,
      N             'LLLLL   AA AA  NN NN  KK  KK  FF     OOOOO  RR  R  DDDD
      O ',14X,' SSS    HH HH     YY    EEEEE  RR  R')
       RETURN
       END
C*     POPULATION FORECASTING        TOTALING SUBROUTINE
       SUBROUTINE TOTAL(POPWM,POPWF,POPNM,POPNF,POPTM,POPTF,TABLE)
       COMMON /TOT/TOTWMC(20),TOTWFC(20),TOTNMC(20),TOTNFC(20),
      1            TOTBMC(20),TOTBFC(20)
       COMMON /RES/RESWMC(20),RESWFC(20),RESNMC(20),RESNFC(20),
      1            RESBMC(20),RESBFC(20)
       DIMENSION POPWM(20),POPWF(20),POPNM(20),POPNF(20),
```

```
     1          POPTM(20),POPTF(20)
      INTEGER YEAR,PAGE,OUTXXX,RACES,RACE,SEX,AGE,CYCLE,BASEYR,
     *        N,INT,TYPE,LASTYR,TABLE,    AGE1,AGE2,M,
     *        M1,M2,M3,INDEXM,INDEXF,YEAR2,PYRMDS
C*
C*    SUM POPULATION
C*
 20   POPWM(20) = 0.
      POPWF(20) = 0.
      POPNM(20) = 0.
      POPNF(20) = 0.
      DO 100 N=1,20
      IF (N .EQ. 20) GO TO 50
      POPWM(20) = POPWM(20) + POPWM(N)
      POPWF(20) = POPWF(20) + POPWF(N)
      POPNM(20) = POPNM(20) + POPNM(N)
      POPNF(20) = POPNF(20) + POPNF(N)
 50   POPTM(N) = POPWM(N) + POPNM(N)
      POPTF(N) = POPWF(N) + POPNF(N)
      IF (TABLE .EQ. 0) GO TO 100
      IF (TABLE .NE. 1 .AND. TABLE .NE. 6) GO TO 80
      TOTWMC(N) = 0.
      TOTWFC(N) = 0.
      TOTNMC(N) = 0.
      TOTNFC(N) = 0.
 80   TOTWMC(N) = TOTWMC(N) + POPWM(N)
      TOTWFC(N) = TOTWFC(N) + POPWF(N)
      TOTNMC(N) = TOTNMC(N) + POPNM(N)
      TOTNFC(N) = TOTNFC(N) + POPNF(N)
      TOTBMC(N) = TOTWMC(N) + TOTNMC(N)
      TOTBFC(N) = TOTWFC(N) + TOTNFC(N)
      IF (TABLE .NE. 9 .AND.TABLE.NE. 10) GO TO 100
      RESWMC(N) = RESWMC(N) + POPWM(N)
      RESWFC(N) = RESWFC(N) + POPWF(N)
      RESNMC(N) = RESNMC(N) + POPNM(N)
      RESNFC(N) = RESNFC(N) + POPNF(N)
      RESBMC(N) = RESWMC(N) + RESNMC(N)
      RESBFC(N) = RESWFC(N) + RESNFC(N)
 100  CONTINUE
      CALL PRINTR(POPWM,POPWF,1,TABLE,1,0)
      CALL PRINTR(POPNM,POPNF,1,TABLE,2,0)
      CALL PRINTR(POPTM,POPTF,1,TABLE,0,0)
C*
C*    PRINT CUMULATIVE   TOTALS
C*
      IF (TABLE .LT. 7 .OR. TABLE .EQ. 8) GO TO 200
      CALL PRINTR(TOTWMC,TOTWFC,1,4,1,0)
      CALL PRINTR(TOTNMC,TOTNFC,1,4,2,0)
      CALL PRINTR(TOTBMC,TOTBFC,1,4,0,0)
 200  IF (TABLE .NE. 10 )                    GO TO 300
C**   CALL PRINTR(RESWMC,RESWFC,1,5,1,0)
C**   CALL PRINTR(RESNMC,RESNFC,1,5,2,0)
C**   CALL PRINTR(RESBMC,RESBFC,1,5,0,0)
 300  RETURN
      END
C*    POPULATION FORECASTING       CHANGE SUBROUTINE
      SUBROUTINE CHANGE(POPM1,POPF1,POPM2,POPF2,TABLE,RACE)
C*    VARIABLES
C*        POPXX  = POPULATION
C*                 X1 - M=MALE,F=FEMALE,T=TOTAL
C*                 X2 - 1=PERIOD 1,2=PERIOD 2
```

```
C*          CHGX    = CHANGE IN POPULATION
C*          PCTX    = PERCENT CHANGE
C*          RACE    = RACE NUMBER
C*
C*          TABLE   = TABLE NUMBER
C*
      COMMON /PER/PCTM(20),PCTF(20),PCTT(20)
      INTEGER YEAR,PAGE,OUTXXX,RACES,RACE,SEX,AGE,CYCLE,BASEYR,
     *        N,INT,TYPE,LASTYR,TABLE,    AGE1,AGE2,M,
     *        M1,M2,M3,INDEXM,INDEXF,YEAR2,PYRMDS
      DIMENSION POPM1(20),POPM2(20),POPF1(20),POPF2(20),
     *          CHGM(20),CHGF(20)
C*    CALCULATE CHANGE
C*
      DO 100 N=1,20
      CHGM(N) = POPM2(N) - POPM1(N)
      CHGF(N) = POPF2(N) - POPF1(N)
      IF (POPM1(N) .EQ. 0. .OR. POPF1(N) .EQ. 0.) GO TO 100
      PCTM(N) = (CHGM(N) / POPM1(N))*100.
      PCTF(N) = (CHGF(N) / POPF1(N))*100.
      PCTT(N) =((CHGM(N) +CHGF(N))/(POPM1(N) + POPF1(N))) *100.
  100 CONTINUE


      CALL PRINTR(CHGM,CHGF,2,TABLE,RACE,IYEAR)
      RETURN
      END
C*    POPULATION FORECASTING        PRINTING SUBROUTINE
      SUBROUTINE PRINTR(POPM,POPF,TYPE,TABLE,RACE,IYEAR)
      REAL LABWMP,LABWFP,LABNMP,LABNFP,LABTBC,LABXXX,NGRI
      REAL INC,MAX
      COMMON /EMP/TER,UNMTBP,LABTBC,LABXXX,BEI,HSI,BSI,AR,NGRI
      COMMON       /PRT/     OUTXXX,SPECL,YEAR,LASTYR,RACES,PAGE,PYRMDS
      COMMON/LAB/LABWMP(20),LABWFP(20),LABNMP(20),LABNFP(20)
      COMMON /PER/PCTM(20),PCTF(20),PCTT(20)
      COMMON /FER/FERWFR(10),FERWFC(10),FERNFR(10),FERNFC(10)
      DIMENSION POPM(20),POPF(20),POPT(20)
      INTEGER YEAR,PAGE,OUTXXX,RACES,RACE,SEX,AGE,CYCLE,BASEYR,
     *        N,INT,TYPE,LASTYR,TABLE,    AGE1,AGE2,M,
     *        M1,M2,M3,INDEXM,INDEXF,YEAR2,PYRMDS
      REAL*8 RNAMES(3),RNAME,TNAMES(33),TNAME1,TNAME2,TNAME3
      INTEGER   PYR(54,121),ONE,TWO,DASH,SPACE,AGES1(10),FILL
      INTEGER   FILL2,MIN1,MIN2,MIN,BAR,XX,LINE
      DATA RNAMES/8H         ,8HWHITE   ,8HNONWHITE/,TNAMES/
     1  8H         ,8H         ,8H         ,8H         ,8H         ,
     1  8H     CUM,8H     BIR,8H      MI,8H     MIL,8H EMPLOY.,
     1  8H RETIRE.,
     1  8H         ,8H         ,8H  TOTAL ,8H         ,8HCUMULATI,
     1  8HULATIVE ,8HTHS AND ,8HLITARY D,8HITARY IN,8H RELATED,
     1  8H RELATED,
     1  8H   TOTAL,8HRESIDENT,8HMILITARY,8H MIGRANT,8HVE TOTAL,
     1  8HRESIDENT,8HSURVIVED,8HEPENDENT,8H-SERVICE,8H MIGRANT,
     1  8H MIGRANT/
      DATA SPACE/1H /,DASH/1H-/,ONE/1H1/,TWO/1H*/
      DATA MIN1/1H=/,MIN2/1H-/,BAR/1HI/,XX/1HX/
      DATA AGES1/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H+/
      DATA HOL1,HOL2/1H+,1H /
C*
C*    CHECK CRITERIA FOR PRINTING TABLE
C*
      IF (OUTXXX .EQ. 1 .AND. TABLE .NE. 0) GO TO 8000
```

```
      IF (TABLE .EQ.4 .AND. OUTXXX .NE. 3) GO TO 8000
      IF (TABLE .EQ.5 .AND. OUTXXX .NE. 3) GO TO 8000
      IF (RACES .NE. 2 .AND. RACE .NE. 0)                    GO TO 8000
C*
      RNAME = RNAMES(RACE + 1)
      IF(TABLE.EQ.11) GO TO 899
      TNAME1 = TNAMES(TABLE +1)
      TNAME2 = TNAMES(TABLE +12)
      TNAME3 = TNAMES(TABLE +23)
  899 CONTINUE
      DO 900 N=1,20
  900 POPT(N) = POPM(N) + POPF(N)
C*
C*    CALCULATE PERCENTAGES (IF CHANGE TABLE, PERCENTS ALREADY CALC.)
C*
      CALL PRINTT(TABLE)
      IF (TYPE .EQ. 2) GO TO 2001
      IF(POPM(20) .EQ. 0.) POPM(20) = .0001
      IF(POPF(20) .EQ. 0.) POPF(20) = .0001
      IF(POPT(20) .EQ. 0.) POPT(20) = .0001
      DO 800 N=1,20
      PCTM(N) =(POPM(N) / POPM(20)) * 100.
      PCTF(N) =(POPF(N) / POPF(20)) * 100.
      PCTT(N) =(POPT(N) / POPT(20)) * 100.
      IF (POPM(20) .EQ. .0001) PCTM(20)= 0.0
      IF (POPF(20) .EQ. .0001) PCTF(20) =0.0
      IF (POPT(20) .EQ. .0001) PCTT(20) =0.0
  800 CONTINUE
      IF(TABLE.EQ.11) GO TO 912
C*
C*    PRINT TITLE
C*
 1001 WRITE (6,901) YEAR,TNAME1,TNAME2,TNAME3,RNAME
  901 FORMAT (' ',    I4,1X,3A8,' POPULATION  ',1A8)
      WRITE (6,911)
  911 FORMAT (1H0,'AGE GROUP',5X,'MALES',5X,'FEMALES',5X,'TOTAL',3X,
     1        'PCT MALE',2X,'PCT FEMALE',2X,'PCT TOTAL')
      GO TO 3000
  912 WRITE(6,914) YEAR,RNAME
      WRITE(6,911)
      GO TO 3000
  914 FORMAT(1X,I4,20X,'LABOR FORCE  ',A8)
C*
 2001 WRITE (6,921) LASTYR,YEAR,TNAME1,TNAME2,TNAME3,RNAME
  921 FORMAT (' ',01X,I4,' - ',I4,1X,3A8,' POPULATION CHANGE  ',1A8)
      WRITE (6,931)
  931 FORMAT (1H0,'AGE GROUP',3X,'MALE CHG',4X,'FEM CHG',4X,'TOT CHG',
     1        2X,'PCT M CHG',2X,'PCT F CHG',2X,'PCT T CHG')
C*
C*    PRINT TABLE
C*
 3000 AGE1 = -5
      DO 3050 N=1,17
      AGE1 = AGE1 +5
      AGE2 = AGE1 +4
      WRITE (6,960) AGE1,AGE2,POPM(N),POPF(N),POPT(N),
     1                        PCTM(N),PCTF(N),PCTT(N)
  960 FORMAT (1X,I4,'-',I2,2X,3(1X,F10.0),3(F9.1,2X))
 3050 CONTINUE
C*
      WRITE (6,965) POPM(18),POPF(18),POPT(18),
```

```
     1                PCTM(18),PCTF(18),PCTT(18)
  965  FORMAT (1X,'   85+     ', 3(1X,F10.0),3(F9.1,2X))
C*
      WRITE (6,970) POPM(20),POPF(20),POPT(20),
     1                PCTM(20),PCTF(20),PCTT(20)
  970  FORMAT (1H0,'  TOTAL   ',3(1X,F10.0),3(F9.1,2X))
      IF (TYPE .EQ. 2 .OR. TABLE .EQ. 4 .OR. TABLE .EQ.5) GO TO 8000
C*
C*    PRINT FERTILITY
C*
      IF (TABLE .EQ. 6 .AND.(RACE .EQ.1 .OR.(RACES.EQ.1.AND.RACE.EQ.0)))
     1   CALL PRINTF(FERWFR,FERWFC,1,1)
      IF (TABLE .EQ. 6 .AND. RACE .EQ. 2)
     1   CALL PRINTF(FERNFR,FERNFC,1,2)
C*
C*    PRINT EMPLOYMENT
C*    PRINT LABOR FORCE PARTICIPATION RATES
C*
      IF (.NOT.(TABLE .EQ. 9 .AND. RACE .EQ. 0)) GO TO 7900
      CALL PRINTT(31)
      WRITE (6,9100) YEAR
 9100 FORMAT (' LABOR FORCE PARTICIPATION RATES, ',I4)
      WRITE (6,9015)
 9015 FORMAT(/,15X,'WHITE     WHITE     NONWHITE     NONWHITE',/,
     *   2X,'AGE GROUP      MALE     FEMALE       MALE         FEMALE',/)
      AGE1 = -5
      DO 1200 N=1,17
      AGE1 = AGE1 + 5
      AGE2 = AGE1 + 4
 1200 WRITE(6,9200)AGE1,AGE2,LABWMP(N),LABWFP(N),LABNMP(N),LABNFP(N)
 9200 FORMAT(3X,I2,'-',I2,8X,F4.3,6X,F4.3,6X,F4.3,8X,F4.3)
      WRITE(6,9201)LABWMP(18),LABWFP(18),LABNMP(18),LABNFP(18)
 9201 FORMAT(3X,'85+',10X,F4.3,6X,F4.3,6X,F4.3,8X,F4.3)
 7900 YEAR2 = YEAR
      IF(TABLE.EQ.11) RETURN
      GO TO 5
 8000 RETURN
C*
C*
C*
C*    POPULATION FORECASTING                PYRAMID WRITER
      ENTRY PYRMID(POPM,POPF,TYPE,TABLE,RACE,IYEAR)
      YEAR2 = IYEAR
 5    IF(PYRMDS .NE. 1) RETURN
      IF((RACES.EQ.1).AND.(RACE.NE.0)) RETURN
      FILL = ONE
      MIN = MIN1
      IF (TYPE .NE. 22) GO TO 35
      FILL = TWO
      MIN = MIN2
      GO TO 38
 35   CALL PRINTT(25)
 38   RNAME = RNAMES(RACE + 1)
      TNAME1 = TNAMES(TABLE +1)
      TNAME2 = TNAMES(TABLE +12)
      TNAME3 = TNAMES(TABLE +23)
      WRITE (6,9710) YEAR2,TNAME1,TNAME2,TNAME3,RNAME,FILL,MIN
 9710 FORMAT (' ',I4,1X,3A8,' POPULATION  ',1A8,'   CHAR.= ',1A1,20X,
     1       '   NEGATIVE CHAR.= ',1A1)
      IF (TYPE  .NE. 21) GO TO 45
      DO 40 N=1,18
```

```
      PCTM(N) = POPM(N)
 40   PCTF(N) = POPF(N)
      RETURN
 45   DO 50 N=1,54
      DO 50 M=1,121
 50   PYR(N,M) = SPACE
C*
      MAX = 0.
      DO 70 N = 1,18
      IF (POPM(N) .GT. MAX) MAX = POPM(N)
      IF (-POPM(N) .GT. MAX) MAX =-POPM(N)
      IF (POPF(N) .GT. MAX) MAX = POPF(N)
      IF (-POPF(N) .GT. MAX) MAX =-POPF(N)
      IF (TYPE .EQ. 1) GO TO 70
      IF (PCTM(N) .GT. MAX) MAX = PCTM(N)
      IF (-PCTM(N) .GT. MAX) MAX =-PCTM(N)
      IF (PCTF(N) .GT. MAX) MAX = PCTF(N)
      IF (-PCTF(N) .GT. MAX) MAX =-PCTF(N)
 70   CONTINUE
      INC = 7500
      IF (MAX .LT. 300000) INC = 5000
      IF (MAX .LT. 150000) INC = 2500
      IF (MAX .LT.  60000) INC = 1000
      IF (MAX .LT.  15000) INC =  250
      IF (MAX .LT.   6000) INC =  100
      IF (MAX .LT.   1500) INC =   25
      GO TO 100
 110  FILL = ONE
      MIN = MIN1
      DO 115 N=1,18
      POPM(N) = PCTM(N)
 115  POPF(N) = PCTF(N)
 100  DO 200 N = 1,18
      M1= 3*N
      FILL2 = FILL
      INDEXM = POPM(N) / INC    + .5
      IF (POPM(N) .GE.0.) GO TO 117
      INDEXM =-POPM(N)  / INC  + .5
      FILL2 = MIN
 117  IF (INDEXM .GT. 60) INDEXM = 60
      M2 = 61-INDEXM
      PYR(M1-1,M2)= FILL2
      IF (PYR(M1-2,M2) .EQ. TWO) GO TO 1173
 1171 PYR(M1-2,M2)= FILL2
 1173 FILL2 = FILL
      INDEXF = POPF(N) / INC    + .5
      IF (POPF(N)  .GE. 0.) GO TO 119
      INDEXF = -POPF(N) / INC  + .5
      FILL2 = MIN
 119  IF (INDEXF .GT. 60) INDEXF = 60
      M2 = INDEXF + 61
      PYR(M1-1,M2)= FILL2
      IF (PYR(M1-2,M2) .EQ. TWO) GO TO  1193
 1191 PYR(M1-2,M2)= FILL2
 1193 M3 = 61- INDEXM
 200  CONTINUE
C*
C*FILL IN HORIZONTAL LINES ON PYRAMID
C*
      LINE = DASH
      IF(TYPE.EQ. 22) LINE = FILL
```

```
      DO 250 N=3,54,3
      DO 250 M1=I,2
      SWITCH = 0.
      DO 250 M2 = 1,60
      M3=M2
      IF (M1 .EQ. 2) M3 = 122 - M2
      IF (N .EQ. 54) GO TO 205
      IF (PYR(N+2,M3) .EQ. FILL .AND. PYR(N-1,M3) .EQ. FILL) GO TO 225
      IF (PYR(N+2,M3) .EQ. FILL) GO TO 210
  205 IF (PYR(N-1,M3) .EQ. FILL) GO TO 210
      IF (SWITCH .EQ. 1.) GO TO 225
      GO TO 250
  210 IF (SWITCH .EQ. 0. ) GO TO 220
      SWITCH = 0.
      GO TO 225
  220 SWITCH = 1.
  225 PYR(N,M3) = LINE
  250 CONTINUE
C*
      IF (TYPE .EQ. 22 .AND. FILL .EQ. TWO) GO TO 110
C*
C*
C*    FILL IN HORIZONTAL LINES OM PYRAMID
C*
      DO 270 N=3,54,3
      DO 260 M1=1,2
      M4=61
      M5=61
      DO 2500 M2= 1,60
      M3 = M2
      IF(M1 .EQ. 2) M3 = 122 - M2
      IF (PYR(N-2,M3) .EQ. SPACE) GO TO 2500
      IF(M5 .EQ. 61) M4 = M3
      IF(M5 .EQ. 61) M6 = PYR(N-2,M3)
      M5 = M3
      M7 = PYR(N-2,M3)
 2500 CONTINUE
      IF(M4 .EQ. 61) GO TO 260
      IF(TYPE.NE.  22) GO TO 2510
      IF(M6 .NE. TWO) GO TO 2510
      IF(M4 .LT. 61 .AND. M4 .LT. M5) M4 = M5
      IF(M4 .GT. 61 .AND. M4 .GT. M5) M4 = M5
 2510 M8 =  M4
      M6 = HOL1
      M7 = HOL2
      IF(M5 .EQ. M4) M6 = M7
      M9 =   M6
 2520 M8 = M8 + 1
      IF(M1 .EQ. 2) M8 = M8 - 2
      IF(M8 .EQ. M5) M9 = M7
      IF(M8 .EQ. M5) GO TO 2550
      PYR(N - 2,M8) = M9
      PYR(N - 1,M8) = M9
      IF(PYR(N,M8) .EQ. SPACE) PYR(N,M8) = M9
 2550 IF(M8 .NE. 61) GO TO 2520
  260 CONTINUE
  270 CONTINUE
C*    LAY OUT SKELETON OF PYRAMID
C*
      DO 400 N=1,54
  400 PYR(N,61) = BAR
```

```
      DO 500 N= 1,9
      M= 6*N-5
      PYR(M,60) = AGES1(N)
      PYR(M+3,60) = AGES1(N)
      PYR(M,61) = AGES1(1)
      PYR(M+3,61) = AGES1(6)
      PYR(M+2,61) = DASH
      PYR(M+5,61) = DASH
  500 CONTINUE
C*
C*    PRINT PYRAMID
C*
      DO 600 N=1,54
      M = 55-N
      WRITE (6,961) (PYR(M,M1),M1=1,121)
  961 FORMAT (10X,121A1)
  600 CONTINUE
      INC = 4.* INC
      WRITE (6,910) INC
  910 FORMAT (' ',9X,49('-'),'MALES',11('-'),'FEMALES',49('-'),
     1                                    /,10X,30('I   '),'I',/,
     1        9X,'15  14  13  12  11  10   9   8   7   6   5   4  ',
     2           ' 3   2   1   0   1   2   3   4   5   6   7   8  ',
     3           ' 9  10  11  12  13  14  15 ',/,9X,'SCALE 1=',F10.0)
      RETURN
      END
C*    POPULATION FORECASTING        FERTILITY PRINTING
      SUBROUTINE PRINTF(FERTR,FERTC,TABLE,RACE)
      COMMON       /PRT/     OUTXXX,SPECL,YEAR,LASTYR,RACES,PAGE,PYRMDS
      DIMENSION FERTR(10),FERTC(10)
      INTEGER YEAR,PAGE,OUTXXX,RACES,RACE,SEX,AGE,CYCLE,BASEYR,
     *        N,INT,TYPE,LASTYR,TABLE,    AGE1,AGE2,M,
     *        M1,M2,M3,INDEXM,INDEXF,YEAR2,PYRMDS
      IF (TABLE .NE. 1) GO TO 8000
      IF (RACE .EQ. 2  .AND. RACES .NE. 2) GO TO 8000
      WRITE (6,900) (FERTR(N),N=4,10)
  900 FORMAT (//,' FERTILITY RATES=  ',6F9.1,'  TFR=  ',F9.1)
C*
      WRITE (6,910) LASTYR,YEAR,(FERTC(N),N=4,10)
  910 FORMAT(1H0,I4,'-',I4,' BABIES= ',6F9.0,'  TOTAL=',F9.0)
 8000 RETURN
      END
C*    POPULATION FORECASTING        LABOR FORCE CALCULATION
      SUBROUTINE LABOR(LABP,POP,LABF,LABC)
      REAL      LABP(20),POP(20),LABF(20),LABC
      LABF(20) = 0.0
      DO 10 I=1,19
      LABF(I) = LABP(I)*POP(I)
      LABF(20) = LABF(20) + LABF(I)
   10 CONTINUE
      LABC = LABF(20)
      RETURN
      END
C*    POPULATION FORECASTING        SURVIVAL AND FERTILITY SUBROUTINE
      SUBROUTINE SURV(POPMI,POPFI,POPMC,POPFC,SURM,SURF,FERR,FERC,BIRMP)
      INTEGER YEAR,PAGE,OUTXXX,RACES,RACE,SEX,AGE,CYCLE,BASEYR,
     *        N,INT,TYPE,LASTYR,TABLE,    AGE1,AGE2,M,
     *        M1,M2,M3,INDEXM,INDEXF,YEAR2,PYRMDS
      DIMENSION POPMI(20),POPFI(20),POPMC(20),POPFC(20),
     1          SURM(20),SURF(20),FERR(10),FERC(10)
C*
```

```
      DO 100 N=1,18
      POPMC(N+1)=POPMI(N) * SURM(N+1)
  100 POPFC(N+1)=POPFI(N) * SURF(N+1)
      POPMC(18) = POPMC(18) + POPMC(19)
      POPFC(18) = POPFC(18) + POPFC(19)
      POPMC(19) = 0.
      POPFC(19) = 0.
C*
C*    FERTILITY CALCULATION
C*
      FERC(10) = 0.
      FERR(10) = 0.
      DO 200 N=4,9
      FERC(N) = FERR(N) *  (POPFI(N)+POPFC(N)+POPFI(N-1)+POPFC(N+1))
     1          / 800.
      FERC(10) = FERC(10) + FERC(N)
      FERR(10) = FERR(10) + 5.*FERR(N)
  200 CONTINUE
      POPMC(1) = FERC(10) * BIRMP* SURM(1)
      POPFC(1) = FERC(10) *(1.-BIRMP)* SURF(1)
      RETURN
      END
C*    POPULATION FORECASTING       SCHOOL ENROLLMENT CALCULATION
      SUBROUTINE SCHOOL (SCHP,POP,SCHE,SCHT)
      DIMENSION SCHP(5,7),SCHE(6,8),POP(20)
      SCHT = 0.0
      DO 5 J=1,8
    5 SCHE(6,J) = 0.0
      DO 10 I=1,6
   10 SCHE(I,8) = 0.0
      DO 20 I=1,5
      DO 20 J=1,7
      SCHE(I,J) = SCHP(I,J)*POP(J)
      SCHT = SCHT + SCHE(I,J)
      SCHE(I,8) = SCHE(I,8) + SCHE(I,J)
      SCHE(6,J) = SCHE(6,J) + SCHE(I,J)
   20 CONTINUE
      DO 30 I=1,5
   30 SCHE(6,8) = SCHE(6,8) + SCHE(I,8)
      RETURN
      END
C*    POPULATION FORECASTING       SCHOOL ENROLLMENT PRINTING
      SUBROUTINE PRINTS(SCHM,SCHF,TYPE)
      COMMON       /PRT/     OUTXXX,SPECL,YEAR,LASTYR,RACES,PAGE,PYRMDS
      INTEGER YEAR,PAGE,OUTXXX,RACES,RACE,SEX,AGE,CYCLE,BASEYR,
     *        N,INT,TYPE,LASTYR,TABLE,     AGE1,AGE2,M,
     *        M1,M2,M3,INDEXM,INDEXF,YEAR2,PYRMDS,I,J,K
      DIMENSION SCHM(6,8),SCHF(6,8),SCHTT(6,8),TS(3,6),
     *          PCTM(6,8),PCTF(6,8),PCTT(6,8)
      REAL*8 WNT(3),AGES(8)
      DATA WNT/8H         ,8HWHITE   ,8HNONWHITE/
      DATA AGES/8H 0- 4   ,8H 5- 9   ,8H10-14   ,8H15-19   ,
     *          8H20-24   ,8H25-29   ,8H30-34   ,8HTOTAL   /
      DATA TS/4HNURS,4HERY ,4H     ,4HKIND,4HERGA,4HRTEN,
     *        4HELEM,4HENTA,4HRY   ,4HHIGH,4H SCH,4HOOL ,
     *        4HCOLL,4HEGE ,4H     ,4HTOTA,4HL   ,4H     /
      ZRACE = WNT(TYPE+1)
      DO 100 I=1,6
      SCHTT(I,8) = SCHM(I,8) + SCHF(I,8)
      DO 100 J=1,8
      PCTM(I,J)=0.
```

```
      PCTF(I,J)=0.
      PCTT(I,J)=0.
      IF (SCHM(I,8).GT.0.) PCTM(I,J) = SCHM(I,J)/SCHM(I,8)*100.
      IF (SCHF(I,8).GT.0.) PCTF(I,J) = SCHF(I,J)/SCHF(I,8)*100.
      SCHTT(I,J) = SCHM(I,J) + SCHF(I,J)
      IF (SCHTT(I,8).GT.0.) PCTT(I,J) = SCHTT(I,J)/SCHTT(I,8)*100
  100 CONTINUE
      DO 500 I=1,6
      IF((I.EQ.1) .OR.(I.EQ.4)) CALL PRINTT(12)
      WRITE(6,900) YEAR,(TS(K,I),K=1,3),ZRACE
      DO 500 J=1,8
      IF(J.EQ.8) WRITE(6,901)
      WRITE(6,910) AGES(J),SCHM(I,J),SCHF(I,J),SCHTT(I,J),
     *             PCTM(I,J),PCTF(I,J),PCTT(I,J)
  500 CONTINUE
      RETURN
  900 FORMAT(1HO,///,1X,I4,14X,'SCHOOL ENROLLMENT - ',3A4,6X,A8,//,
     *       1X,'AGE GROUP          MALES     FEMALES     TOTAL',
     *          '   PCT MALE  PCT FEMALE  PCT TOTAL',/)
  901 FORMAT(' ')
  910 FORMAT(3X,A8,6X,F8.0,4X,F8.0,3X,F8.0,4X,F5.1,2(6X,F5.1))
      END
C*    POPULATION FORECASTING       GROUP QUARTERS CALCULATIONS
      SUBROUTINE GROUPQ(GQ1,GQP1,POP1,GQ2,GQP2,POP2)
      INTEGER YEAR,PAGE,OUTXXX,RACES,RACE,SEX,AGE,CYCLE,BASEYR,
     *        N,INT,TYPE,LASTYR,TABLE,    AGE1,AGE2,M,
     *        M1,M2,M3,INDEXM,INDEXF,YEAR2,PYRMDS,I,J,K
      DIMENSION GQ1(11),POP1(20),GQP1(10),
     *          GQ2(11),POP2(20),GQP2(10)
C*
C*    TABLE = 30 IS HEADS OF HOUSEHOLDS
C*
      GQ1(11) = 0.0
      GQ2(11) = 0.0
      GQ1(1) = GQP1(1)*(POP1(1) + POP1(2) + POP1(3))
      GQ2(1) = GQP2(1)*(POP2(1) + POP2(2) + POP2(3))
      DO 100 I=2,5
      GQ1(I) = GQP1(I)*POP1(I+2)
  100 GQ2(I) = GQP2(I)*POP2(I+2)
      DO 200 I=6,9
      GQ1(I) = GQP1(I)*(POP1(2*I-4) + POP1(2*I-3))
  200 GQ2(I) = GQP2(I)*(POP2(2*I-4) + POP2(2*I-3))
      GQ1(10) = GQP1(10)*(POP1(16) + POP1(17) + POP1(18))
      GQ2(10) = GQP2(10)*(POP2(16) + POP2(17) + POP2(18))
      DO 300 I=1,10
      GQ1(11) = GQ1(11) + GQ1(I)
  300 GQ2(11) = GQ2(11) + GQ2(I)
      RETURN
      END
C*    POPULATION FORECASTING       GROUP QUARTERS PRINTING
      SUBROUTINE PRINTG(GQM,GQF,KIND,RACE)
      COMMON       /PRT/     OUTXXX,SPECL,YEAR,LASTYR,RACES,PAGE,PYRMDS
      INTEGER YEAR,PAGE,OUTXXX,RACES,RACE,SEX,AGE,CYCLE,BASEYR,
     *        N,INT,TYPE,LASTYR,TABLE,    AGE1,AGE2,M,I,J,
     *        M1,M2,M3,INDEXM,INDEXF,YEAR2,PYRMDS,KIND
      DIMENSION GQM(11),GQF(11),GQT(11),KGQ(3,4),
     *          PCTM(11),PCTF(11),PCTT(11)
      REAL*8    WNT(3),AGES(11)
      DATA WNT/8H        ,8HWHITE   ,8HNONWHITE/
      DATA KGQ/4HMILI,4HTARY,4H    ,4HCOLL,4HEGE ,4HDORM,
     *         4HOTHE,4HR   ,4H    ,4HTOTA,4HL   ,4H    /
```

```
      DATA AGES/5H 0-14,5H15-19,5H20-24,5H25-29,5H30-34,
     *                  5H35-44,5H45-54,5H55-64,5H65-74,5H75+   ,5HTOTAL/
C*
C*    KIND = 5 IS HEADS OF HOUSEHOLDS
C*    KIND = 6 IS HOUSEHOLD POPULATION
C*
      ZRACE = WNT(RACE+1)
      DO 100 I=1,11
  100 GQT(I) = GQM(I) + GQF(I)
      IF((RACES.NE.2).AND.(RACE.NE.0)) GO TO 1000
      DO 200 I=1,11
      PCTM(I) = GQM(I)/GQM(11)*100.
      PCTT(I) = GQT(I)/GQT(11)*100.
      IF(KIND.NE.1) GO TO 199
      PCTF(I) = 0.0
      GO TO 200
  199 PCTF(I) = GQF(I)/GQF(11)*100.
  200 CONTINUE
      IF(RACES.NE.2) GO TO 210
      IF(RACE.NE.2) CALL PRINTT(14)
      IF(RACE.EQ.2) WRITE(6,920)
      GO TO 215
  210 IF((KIND.EQ.2).OR.(KIND.EQ.4)) GO TO 220
      CALL PRINTT(14)
  215 IF(KIND.EQ.5) GO TO 250
      IF(KIND.EQ.6) GO TO 240
      GO TO 230
  220 WRITE(6,920)
  230 WRITE(6,900) YEAR,(KGQ(J,KIND),J=1,3),ZRACE
      WRITE(6,902)
      GO TO 260
  240 WRITE(6,905) YEAR,ZRACE
      WRITE(6,902)
      GO TO 260
  250 WRITE(6,901) YEAR,ZRACE
      WRITE(6,902)
  260 DO 300 I=1,11
      IF(I.EQ.11) WRITE(6,903)
      WRITE(6,910) AGES(I),GQM(I),GQF(I),GQT(I),
     *             PCTM(I),PCTF(I),PCTT(I)
  300 CONTINUE
 1000 RETURN
  900 FORMAT(1H0,///,1X,I4,15X,'GROUP QUARTERS - ',3A4,2X,A8,//)
  901 FORMAT(1H0,///,1X,I4,14X,'HEADS OF HOUSEHOLDS   ',2X,A8,//)
  902 FORMAT(1X,'AGE GROUP          MALES     FEMALES      TOTAL',
     *       '    PCT MALE  PCT FEMALE  PCT TOTAL',/)
  903 FORMAT('    ')
  905 FORMAT(1H0,///,1X,I4,14X,'HOUSEHOLD POPULATION ',2X,A8,//)
  910 FORMAT(3X,A8,6X,F8.0,4X,F8.0,3X,F8.0,4X,F5.1,2(6X,F5.1))
  920 FORMAT(//////)
      END
C*    POPULATION FORECASTING                TITLE PRINTING
      SUBROUTINE PRINTT(TABLE)
      COMMON       /PRT/     OUTXXX,SPECL,YEAR,LASTYR,RACES,PAGE,PYRMDS
      INTEGER YEAR,PAGE,OUTXXX,RACES,RACE,SEX,AGE,CYCLE,BASEYR,
     *        N,INT,TYPE,LASTYR,TABLE,     AGE1,AGE2,M,
     *        M1,M2,M3,INDEXM,INDEXF,YEAR2,PYRMDS
      WRITE(6,900)     PAGE
  900 FORMAT (1H1,'POPULATION AND EMPLOYMENT FORECAST',10X,
     120X,'PAGE',I8)
      PAGE = PAGE + 1
```

```
RETURN
END
```

FERT: The Fertility Data Generator

Purpose

FERT computes age specific fertility rates in response to specification of a few parameters and initial values. Final age specific fertility rates are computed, and (by linear interpolation) the other cycle age specific fertility rates are computed. The cycle values are merged into other APPLE data sets to complete the cohort survival model.

Procedure

In general, the end year age specific fertility rate is computed as follows:

$$\text{age specific fertility rate (i)} = [\text{projected completed fertility}^{(a)} \times \tfrac{1}{5}] \times {}^{(b)}(\text{proportion of births born to women in a specific age group})^{(c)} \quad (1)$$

Each step is explained below indexed by the superscript.

(a) Projected completed fertility is the estimate of the end year completed fertility rate, expressed as the number of children per woman at the completion of her childbearing years.

(b) The relationship of age specific fertility rate to the completed fertility rate is:

$$CFR = \left[\sum_{i=1} ASBR\ (i)\right] * 5 \quad (2)$$

The above relationship is obvious since completed family size is accomplished by considering the habits of all chilbearing women. To each cohort is attributed a fertility rate, an average representing a population of women aged (10+5i) to (14+5i). When aggregating these rates to describe completed fertility each cohort must allow a contribution to the sum for each of its year members. This accounts for the need to multiply the sum of such rates by 5.

(c) Multiplication by the proportion of births in each cohort allocates the total number of births according to the input pattern.

Unlike labor force participation rate changes and survival rate changes, changes in fertility patterns assume varying end years. To assume the projected completed fertility rate, fertility "habits" will change with the first new cohort to enter the system. This new cohort in cycle 1 was below childbearing age in cycle 0. In the 30 years it requires for this cohort to pass through the system at each cycle the cohort will take on the end year age specific fertility rate.

If for example, the base year is 1970 and the end year of the APPLE run is 2000, those women between 10 and 14 in 1970 will have entered the childbearing years by 1975 and compose cohort 1. This new cohort will assume the end year fertility rate for this age group. In 1980, this 20-24 aged cohort will assume end year fertility rates of that cohort. The group continues to move through the system, at each cycle it adopts the appropriate end year fertility rate. Cohort 1 adopts end year fertility rates at base year + one cycle, cohort 2 at base + 2 cycles, cohort 3 at base + 3 cycles, etc.

Using the final year age specific fertility rates and the timing of the final years by cohort, simple linear interpolation methods are used to obtain the intermediate rates. Cohorts already in the system by the base year assume these calculated values. These rates are mid cycle rates, however, so the value for (base year + cycle/2) is actually calculated to represent the simple average [rate for base year + (I-1) *cycle + rate for base year + I *cycle] /2. For example, to determine the rate for cohort 3

in base + cycle year, the following is the calculation: (base=1970), cycle =5.

$$\frac{1975-1970-2.5}{1985-1970} * (\text{base year rate} - \text{end year rate}) + \text{base year rate.} \qquad (3)$$

The fertility rate generator is designed to incorporate additional information into the calculation of age specific fertility rates. A test run of the Version one program, when compared to actual data, revealed the limitations resultant from the use of linear interpolation. Instead of applying curvilinear approximation routines to additional data input, it was decided that the program could still provide useful estimates employing linear interpolation having exercised careful study in the choice of a base year fertility rate.

The main program in FERT version two remains similar to the Version one FERT program. If an additional data flag is read, control is transferred to the sub-routine ACTUAL to process the updating information.

ACTUAL reads each of the sets of data (one set for each five year cycle). The cycle averages and mid-point year are computed. If data are unavailable for certain years, the zero values are excluded from the averaging computations.

For each cycle for which there are at least 3 data updates, the mid-cycle age specific fertility rates are computed and written to the output file. Note that these values are not results of an interpolation algorithm but rather are simple averages of cycle updates. When no more cycles can be processed in this way, a new "base" year must be selected. This year, and its accompanying period fertility rate statistic, will serve as the starting point for the interpolation. The base year chosen is that beginning of a cycle for which there are two or fewer data points provided.

Having developed a new set of "base" year age specific fertility rates, control returns to main to generate the appropriate interpolated values.

FIGURE 4

FERT PROGRAM FLOW

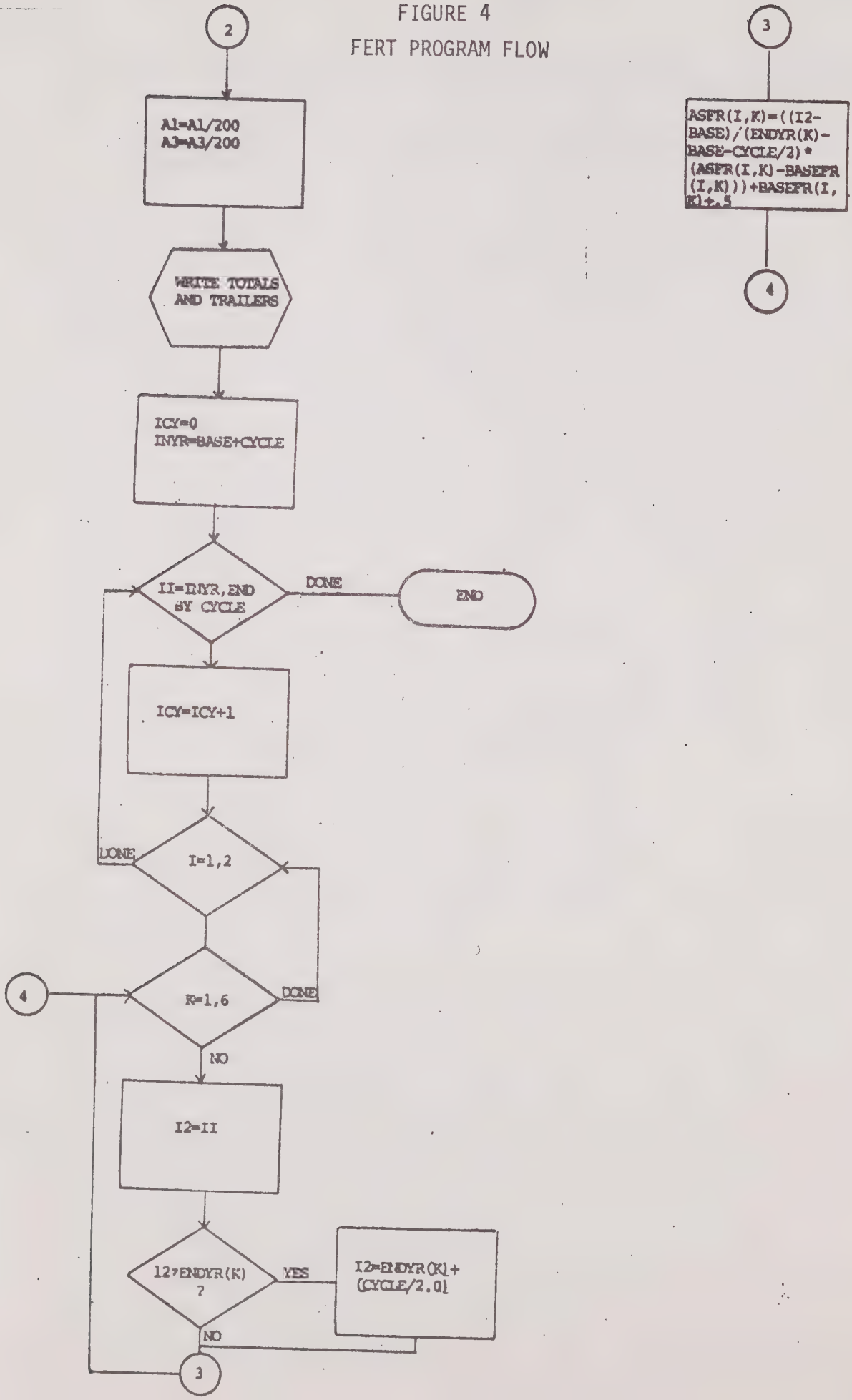
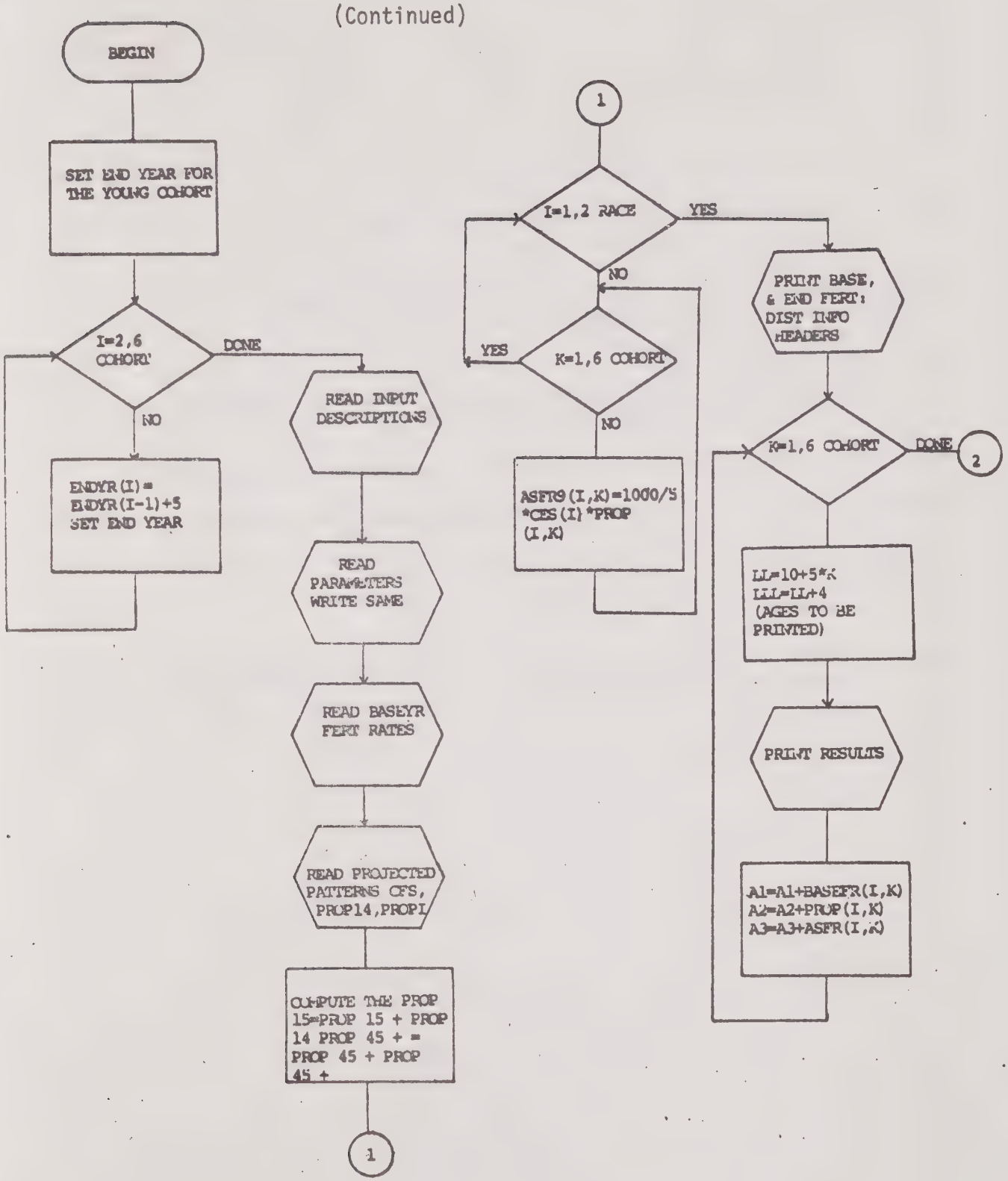


FIGURE 4

(Continued)



Dictionary of Variables

ASFR -calculated age specific fertility rates
an array of dimension (2,6)

ASFR0 -base year age specific fertility rates
an array of dimension (2,6)

ASFR9 -end year age specific fertility rates
an array of dimension (2,6)

BASEFR -base year 5-year age specific fertility rates
an array of dimension (2,6)
read from input data

CFS -completed family size (completed fertility rate) does not count adult family members, indexed by race

PROP14 -proportion of births born to women under 15 years of age

PROP45 -proportion of births born to women over 45 years of age

PROP -proportion of births by cohort
an array of dimension (2,6)
read from input data

ENDYR -year at which each cohort of women assumes endyear fertility "habits"
a vector of dimension (6)

FERT Program

Input Description

The generator requires base year and end year dates and cycle length. In addition, base year age specific fertility rates and a projected timing pattern must be supplied. Base year age specific fertility rates represent births per 1000 women. To compute the completed fertility rates these age specific rates represent, the age specific rates are added, divided by five, and then divided by 1000, to obtain number of children per woman. The projected timing pattern indicates what proportion of births occur in each cohort throughout the child bearing years. The data are four digit decimals.

Below are the specification for the necessary input to FERT.

CARD	COLUMNS	DESCRIPTION
1		TITLE (20A4)
	1-80	Any title
2		PARAMETERS (2I4, 2I1)
	1-4	Base year
	5-8	End year of APPLE run
	9	Cycle length
	10	Number of races

3		BASE YEAR FERTILITY (6F4.4)
	1-4	15-19 base year fertility rate
	5-8	20-24 base year fertility rate
	9-12	25-29 base year fertility rate
	13-16	30-34 base year fertility rate
	17-20	35-39 base year fertility rate
	21-24	40-44 base year fertility rate
4		BASE YEAR FERTILITY, for race 2
		Same as Card No. 3
5		END YEAR FERTILITY (F5.2, 5X, 8F4.4)
	1-5	The End year projected completed fertility rate (right justified)
	6-10	Blank
	11-14	% births born to women under 14
	15-18	% births born to women 15-19
	19-22	% births born to women 20-24
	23-26	% births born to women 25-29
	27-30	% births born to women 30-34
	31-34	% births born to women 35-39
	35-38	% births born to women 40-44
	39-42	% births born to women over 45
6		END YEAR FERTILITY, for race 2
		Same as Card No. 5

FERT Program (Version two)

Input Description

Additional data, in the form of the completed family size (CFS) statistics, are optional as input. If such data are available, the parameter record item "IDATA" will indicate the number of years for which additional data are provided. The format of the parameter card is similar to that used in version one with that one exception:

CARD	COLUMNS	DESCRIPTION
2		Parameters (2I4,2I1,I2)
	1-4	Base Year
	5-8	End year of APPLE run
	9	Cycle length
	10	Number of races
	11-12	Number of additional years of data (IDATA)

Base year data, the timing pattern and the end year completed fertility rate are input as in version one. However, if columns 11 and 12 of the parameter card are greater than zero, additional data records are required. Each record lists the first run of a cycle and the CFS values for that cycle.

CARD	COLUMNS	DESCRIPTION (I4,5(F5.2))
7	1-4	The initial year for cycle of data
	5-9	Data for initial year for cycle of data
	10-14	Data for second year in cycle
	15-19	Data for third year in cycle
	20-24	Data for fourth year in cycle
	25-29	Data for fifth year in cycle
8		Same as Card 7
etc.		As required

Output Description

Because version two makes an important "decision" in computing a new base year to be used in the interpolation computations, further reporting of the program's intermediate results is necessary. The report indicates new base year and accompanying rates, and intermediate interpolated age specific fertility rates and the completed family size.

The data file is generated in the same format as is found in version one.

TECHNICAL SPECIFICATIONS

FERT reads file 5, SYSIN, and writes the report to file 6, SYSOUT. The output data file is written to file 10. Below is the JCL (Job Control Language) necessary to generate these files when executing FERT:

```
//FT05F001  DD DDNAME=SYSIN
//FT06F001  DD SYSOUT=A
//FT10F001  DD DSN=name,UNIT=unit type,SPACE=(TRK,(3,1),RLSE),
//          VOL=SER=volname,DISP=(,CATLG),DCB=(LRECL=80,BLKSIZE=3520,
//          RECFM=FB)
//SYSIN     DD *
  data
/*
```

Timing and core requirements for execution of FERT are:

CPU TIME	.009
REGION USED	38K

```
C
C     PROGRAM GENERATES 5 YEAR AGE SPECIFIC FERTILITY RATES BASED ON
C     TIMING PATTERN OF CHILD BEARING AND COMPLETED FAMILY SIZE
      REAL REM(20),BASEFR(2,6),CFS(2),PROP(2,6),PROP14,PROP45,
     1ASFR (2,6)
      REAL ASFR9 (2,6)
      INTEGER ENDYR (6),ASFRC (2,6),IASFR(2,6)
      INTEGER BASE,END,CYCLE,RACES
C     ASFRC          -    CALCULATED AGE SPECIFIC FERTILITY RATES
C     ASFRO          -    BASE YEAR AGE SPECIFIC FERTILITY RATES
C     ASFR 9         -    FINAL AGE SPECIFIC FERTILITY RATES
C     BASEFR         -    BASE YEAR 5-YEAR AGE SPECIFIC FERTILITY RATE
C     CFS            -    COMPLETED FAMILY SIZE
C     PROP 14        -    PROPORTION BIRTHS BORN TO WOMEN UNDER 14
C     PROP 45        -    PROPORTION BIRTHS BORN TO WOMEN OVER 45
C     PROP(I,K)      -    PROPORTION BIRTHS BORN TO WOMEN OF RACE I
C                    -    AND IN AGE COHORT K
C     ENDYR          -    YEAR AT WHICH THIS COHORT ADOPTS NEW HABITS
C     SET END YEAR FOR INTERPOLATION
C     RATES ARE EACH INTERPOLATED BETWEEN BASE YEAR AND END YEAR FOR
C     THE AGE SPECIFIC RATE
C     EACH OF THESE LOOPS USES RACES AS A CONTROL
C     IF RACES > 2, THE DELIMITERS MUST BE CHANGED
C     ACCORDINGLY
C     READ INPUT DESCRIPTION
      READ (5,901 ) REM
901   FORMAT (20A4)
C     READ  PARAMETER CARD
      READ (5,902) BASE,END,CYCLE,RACES
902   FORMAT (2I4,9X,I1,3X,I2)
      ENDYR (1) = BASE + 5
      DO 50 I = 2,6
      END YR (I) = END YR (I-1)+5
      WRITE (6,999) I,ENDYR(I)
999   FORMAT (//,'  END YEAR CALCULATIONS--',10X,
     X'END YEAR FOR COHORT ',I1,' = ',2X,I4,10X)
50    CONTINUE
C     PRINT OUT PARAMETERS
950   FORMAT('1 BASE YEAR FERTILITY RATES AND FINAL FERTILITY RATES
C     READ BASE YEAR FERTILITY RATES
      READ (5,910) ((BASEFR(I,J),J=1,6),I=1,2)
910   FORMAT (6 F4.4,/,6F4.4)
C     READ PROJECTED PATTERNS OF FERTILITY, ADD AGES UNDER 15 AND O
      READ (5,920) CFS (1),PROP14,(PROP (1,K),K=1,6),PROP45
920   FORMAT ( F5.2,5X,8 F4.4)
      PROP ( 1,1 ) = PROP (1,1 ) + PROP14
      PROP ( 1,6 ) = PROP ( 1,6 ) + PROP45
      READ (5,920 ) CFS (2), PROP14, (PROP (2,K),K=1,6),PROP45
      PROP (2,1) = PROP (2,1) + PROP45
      PROP (2,6) = PROP (2,6) + PROP45
C     CALCULATE FINAL AGE SPECIFIC FERTILITY RATES
C     TO CALCULATE THE AGE SPECIFIC FERTILITY RATE
C     THE COMPLETED FAMILY SIZE IS DIVIDED BY 5
C     AND THEN EACH COHORT'S % OF BIRTHS IS APPLIED TO
```

```
C     THIS FIGURE
      DO 200 I = 1,2
      DO 200 K = 1,6
      ASFR9 (I,K) =.2  * CFS (I) * PROP (I,K)
200   CONTINUE
C     PRINT OUT BASE YEAR , FINAL RATES
      DO 300 I=1,2
      IF (I.EQ.2 .AND. RACES .EQ.1) GO TO 300
      WRITE (6,950)
      IF (I .EQ.1 .AND. RACES .EQ. 2) WRITE (6,960)
960   FORMAT (' WHITE FEMALE',//)
      IF (I .EQ. 2 .AND. RACES .EQ. 2) WRITE (6,970)
970   FORMAT (' NONWHITE FEMALE',//)
      WRITE (6,975) BASE
975   FORMAT ('  AGE ',10X,I5,10X,'FINAL',10X,'FINAL',/
     X          16X, 'FERT.',10X,'BIRTH',10X,'FERT.',/
     X          16X,'RATES',10X,'DIST.',10X,'RATES',//)
      A1 = 0.
      A2 = 0.
      A3 = 0.
      DO 280 K = 1,6
      LL = 10 + 5 * K
      LLL = LL + 4
      WRITE (6,980) LL,LLL,BASEFR(I,K), PROP(I,K),ASFR9(I,K)
      A1 = A1 + BASEFR (I,K)
      A2 = A2 + PROP (I,K)
      A3 = A3 + ASFR9 (I,K)
980   FORMAT (1X,I2,'-',I2,10X,F5.4,10X,F5.4,10X,F5.4)
280   CONTINUE
      A1 = A1*5.
      A3 = A3*5.
      WRITE (6,985) A2,A1,A3
985   FORMAT (//,' SUM',26X,F6.3,/,' CFS',11X,F5.2,25X,F5.2)
300   CONTINUE
      ICY = 0
      INYR = BASE + CYCLE
      DO 500 II = INYR,END,CYCLE
      ICY = ICY + 1
C     GENERATE AGE SPECIFIC FERTILITY RATES FOR EACH CYCLE
C     USES LINEAR INTERPOLATION BETWEEN BASE YEAR RATES AND FINAL R
C     END YEAR FOR EACH RATE IS DIFFERENT
      DO 400 I = 1,2
      DO 380 K = 1,6
      I2 = II
      IF (I2 .GE. ENDYR (K)) GO TO 370
      ASFR(I,K)=((I2-BASE-CYCLE/2.)/(ENDYR(K)-BASE))*(ASFR9(I,K)
     --BASEFR(I,K))+BASEFR(I,K)+.00005
      GO TO 380
370   ASFR(I,K) = ASFR9(I,K)
380   CONTINUE
      I4 = I + 8
      DO 222 KK=1,2
      DO 222 JJ=1,6
      IASFR(KK,JJ)=IFIX(10000.*ASFR(KK,JJ))
222   CONTINUE
      WRITE (10,990) (IASFR(I,KK),KK=1,6),ICY,I4
```

```
990     FORMAT (6I4,51X,I2,I2,' ')
C       THIS IS A DUMMY WRITE STATEMENT TO TEST THE PROGRAM
        IF((I .EQ. 2) .AND.(RACES .EQ. 1)) GO TO 400
        WRITE (6,111)
111     FORMAT (//,'  INTERMEDIATE BIRTH RATES',//,38X,'YEAR',
       X8X,'15-19',5X,'20-24',5X,'25-29',5X,'30-34',5X,'35-39',
       X5X,'40-44', 5X)
        WRITE (6,115) II, (ASFR(I,KK),KK=1,6)
115     FORMAT (//,38X,I4,8X,6(F5.4,5X))
400     CONTINUE
500     CONTINUE
        ENDFILE 10
        REWIND 10
        STOP
        END
```

```
C
C     PROGRAM GENERATES 5 YEAR AGE SPECIFIC FERTILITY RATES
C     BASED ON TIMING PATTERN OF CHILD BEARING AND COMPLETED
C     FERTILITY RATE
C
C     WRITTEN BY NANCY G. SHYER, OCTOBER, 1975
C
      COMMON REM,BASEFR,CFS,PROP,PROP14,PROP45,ASFR,ASFR9,CFSSUM,AVYR,
     *ACFS,AVCFS,ADJUST,BASCFS,CFERT,ENDYR,IASFR,BASE,BASEY,END,CYCLE,
     *RACES,IDATA,IYRSUM,IYEAR,IMATCH
      REAL REM(20),BASEFR(2,6),CFS(2),PROP(2,6),PROP14,PROP45,
     1ASFR(2,6,10),ASFR9(2,6),CFSSUM(10),AVYR(10),ACFS(10,5),
     2AVCFS(10),ADJUST,BASCFS,CFERT(10),JOY(2)
      INTEGER ENDYR(6),IASFR(2,6,10),BASE,BASEY,END,CYCLE,RACES,
     1IDATA,IYRSUM(10),IYEAR(10),IMATCH
C     ASFR0              BASE YEAR AGE SPECIFIC FERTILITY RATES
C     ASFR9              FINAL AGE SPECIFIC FERTILITY RATES
C     BASEFR             BASE YEAR 5-YR AGE SPECIFIC FERTILITY RATES
C     CFS                COMPLETED FERTILITY RATE
C     PROP14             PROPORTION BIRTHS BORN TO WOMEN UNDER 14
C     PROP(I,K)          PROPORTION BIRTHS BORN TO WOMEN OF RACE I
C                        AND IN AGE COHORT K
C     PROP45             PROPORTION BIRTHS BORN TO WOMEN OVER 45
C     ENDYR              YEAR AT WHICH THIS COHORT ADOPTS NEW HABITS
C     SET END YEAR FOR INTERPOLATION
C     RATES ARE EACH INTERPOLATED BETWEEN BASE YEAR AND END YEAR FOR
C     THE AGE SPECIFIC RATE
C     EACH OF THESE LOOPS USES RACES AS A CONTROL
C     IF RACES > 2 THE DELIMITERS MIST BE CHANGED
C     ACCORDINGLY
C     READ INPUT DESCRIPTION
      READ (5,901) REM
901   FORMAT (20A4)
C     READ PARAMETER  CARD
      READ (5,902) BASE,END,CYCLE,RACES,IDATA
902   FORMAT (2I4,2I1,I2)
      ENDYR(1) = END - (CYCLE*6)
      DO 50 I=2,6
      ENDYR(I) = ENDYR(I-1) + 5
      WRITE (6,999) I,ENDYR(I)
999   FORMAT (//,'  END YEAR CALCULATIONS--',10X,
     X'END YEAR FOR COHORT ',I1,' = ',2X,I4,10X)
50    CONTINUE
C     PRINT OUT PARAMETERS
      WRITE (6,899) BASE,END,CYCLE,IDATA
899   FORMAT ('1  FERTILITY RATE GENERATOR  ',//,' BASE YEAR = ',
     XI4,/,' END YEAR = ',I4,/' CYCLE LENGTH = ',I1,
     X/,' NUMBER OF UPDATES = ',I2)
950   FORMAT('1 BASE YEAR FERTILITY RATES AND FINAL FERTILITY RATES',//)
C     READ BASE YEAR FERTILITY RATES
      READ (5,910) ((BASEFR(I,J),J=1,6),I=1,2)
910   FORMAT (6F4.4,/,6F4.4)
C     READ PROJECTED PATTERNS OF FERTILITY, ADD AGES UNDER 15 AND OVER 4
      READ (5,920) CFS (1),PROP14,(PROP (1,K),K=1,6),PROP45
920   FORMAT ( F5.2,5X,8F4.4)
      PROP ( 1,1 ) = PROP (1,1 ) + PROP14
      PROP ( 1,6 ) = PROP ( 1,6 ) + PROP45
```

```
      READ (5,920 ) CFS (2), PROP14, (PROP (2,K),K=1,6),PROP45
      PROP (2,1) = PROP (2,1) + PROP45
      PROP (2,6) = PROP (2,6) + PROP45
C     CALCULATE FINAL AGE SPECIFIC FERTILITY RATES
C     TO CALCULATE THE AGE SPECIFIC FERTILITY RATE
C     THE COMPLETED FAMILY SIZE IS DIVIDED BY 5
C     AND THEN EACH COHORT'S % OF BIRTHS IS APPLIED TO
C     THIS FIGURE
      DO 200 I = 1,2
      DO 200 K = 1,6
      ASFR9 (I,K) =.2 * CFS (I) * PROP (I,K)
200   CONTINUE
C     PRINT OUT BSE YEAR , FINAL RATES
      DO 300 I=1,2
      IF (I.EQ.2 .AND. RACES .EQ.1) GO TO 300
      WRITE (6,950)
      IF (I .EQ.1 .AND. RACES .EQ. 2) WRITE (6,960)
960   FORMAT (' WHITE FEMALE',//)
      IF (I .EQ. 2 .AND. RACES .EQ. 2) WRITE (6,970)
970   FORMAT (' NONWHITE FEMALE',//)
      WRITE (6,975) BASE
975   FORMAT ('  AGE ',10X,I5,10X,'FINAL',10X,'FINAL',/
     X          16X, 'FERT.',10X,'BIRTH',10X,'FERT.',/
     X          16X,'RATES',10X,'DIST.',10X,'RATES',//)
      A1 = 0.
      A2 = 0.
      A3 = 0.
      DO 280 K = 1,6
      LL = 10 + 5 * K
      LLL = LL + 4
      WRITE (6,980) LL,LLL,BASEFR(I,K), PROP(I,K),ASFR9(I,K)
      A1 = A1 + BASEFR (I,K)
      A2 = A2 + PROP (I,K)
      A3 = A3 + ASFR9 (I,K)
980   FORMAT (1X,I2,'-',I2,10X,F5.4,10X,F5.4,10X,F5.4)
280   CONTINUE
      A1 = A1*5.
      A3 = A3*5.
      WRITE (6,985) A2,A1,A3
985   FORMAT (//,' SUM',26X,F6.3,/,' CFS',11X,F5.2,25X,F5.2)
300   CONTINUE
      ICY = 0
      MU = 0
      BASEY = BASE
      IF (IDATA .GT. 0) CALL ACTUAL
      ML = MU + 1
      MU = (((END - BASE)/CYCLE)-ML) + 1
      DO 500 II = ML,MU
      ICY = ICY + 1
C     GENERATE AGE SPECIFIC FERTILITY RATES FOR EACH CYCLE
C     USES LINEAR INTERPOLATION BETWEEN BASE YEAR RATES AND FINAL RATES
C     END YEAR FOR EACH RATE IS DIFFERENT
      DO 400 I = 1,2
      DO 380 K = 1,6
      I2 = (II*CYCLE) + BASE
      IF (I2 .GE. ENDYR (K)) GO TO 370
      ASFR(I,K,II)=((I2-BASEY-CYCLE/2.)/(ENDYR(K)-BASEY))*(ASFR9(I,K)
     --BASEFR(I,K))+BASEFR(I,K)+.00005
      GO TO 380
370   ASFR(I,K,II) = ASFR9(I,K)
```

```
380   CONTINUE
      I4 = I + 8
      JOY(I) = 0.
      DO 222 JJ=1,6
      JOY(I) = JOY(I) + ASFR(I,JJ,II)
      IASFR(I,JJ,II)=IFIX(10000.*ASFR(I,JJ,II))
222   CONTINUE
      JOY(I) = JOY(I) * 5.
      WRITE (10,990) (IASFR(I,KK,II),KK=1,6),ICY,I4
990   FORMAT (6I4,51X,I2,I2,' ')
C     THE FOLLOWING CAUSES INTERVAL VALUES TO BE PRINTED.
      IF ((I .EQ. 2) .AND.(RACES .EQ. 1)) GO TO 400
      WRITE (6,111)
111   FORMAT (//,'  INTERMEDIATE BIRTH RATES',//,38X,'YEAR',
     X8X,'15-19',5X,'20-24',5X,'25-29',5X,'30-34',5X,'35-39',
     X5X,'40-44', 5X,4X,'CFS')
      WRITE (6,115) I2, (ASFR(I,KO,II),KO=1,6) ,JOY(I)
115   FORMAT (//,38X,I4,8X,6(F5.4,5X),4X,F5.2)
400   CONTINUE
500   CONTINUE
      STOP
      END
      SUBROUTINE ACTUAL
      COMMON REM,BASEFR,CFS,PROP,PROP14,PROP45,ASFR,ASFR9,CFSSUM,AVYR,
     *ACFS,AVCFS,ADJUST,BASCFS,CFERT,ENDYR,IASFR,BASE,BASEY,END,CYCLE,
     *RACES,IDATA,IYRSUM,IYEAR,IMATCH
      REAL REM(20),BASEFR(2,6),CFS(2),PROP(2,6),PROP14,PROP45,
     1ASFR(2,6,10),ASFR9(2,6),CFSSUM(10),AVYR(10),ACFS(10,5),
     1AVCFS(10),ADJUST,BASCFS,CFERT(10),JOY(2)
      INTEGER ENDYR(6),IASFR(2,6,10),BASE,BASEY,END,CYCLE,RACES,
     *IDATA,IYRSUM(10),IYEAR(10),IMATCH
C     THIS SUBROUTINE PROCESSES THE ACTUAL FERTILITY DATA
C     AND ALLOWS IT TO BE INCLUDED IN THE INTERPOLATIONS.
      DO 5 I = 1,10
      IYRSUM(I) = 0
5     CFSSUM(I) = 0
      JL = 0
      K = 1
4     READ (5,100,END=6) IYEAR(K),(ACFS(K,J),J=1,5)
      WRITE  (6,101) IYEAR(K),(ACFS(K,J),J=1,5)
100   FORMAT  (I4,5F5.2)
101   FORMAT(1X,I4,5F5.2,'******************')
      DO 3 I = 1,5
      IF (ACFS(K,I) .EQ. 0) GO TO 3
      CFSSUM(K) = CFSSUM(K) + ACFS(K,I)
      IYRSUM(K) = IYRSUM(K) + (I-1) + IYEAR(K)
      JL = JL + 1
3     CONTINUE
      AVCFS(K) = CFSSUM(K)/JL
      AVYR(K) = IYRSUM(K)/JL
      WRITE (6,12) AVCFS(K),AVYR(K)
12    FORMAT ('  AVCFS = ', F5.2, '  AVYR = ', F7.0)
      K = K + 1
      IYRSUM(K) = 0
      CFSSUM(K) = 0
      JL = 0
      GO TO 4
6     K = K - 1
      DO 7 I = 1,5
7     IF (ACFS(K,I) .NE. 0) LAST = I
```

```
      IF (LAST.GT.3) BASEY=IYEAR(K)+CYCLE
      IF (LAST .LT. 3) BASEY = IYEAR(K)
C     NOW BASE YEAR AGE SPECIFIC FERTILITY RATES MUST BE COMPUTED.
C     LINEAR INTERPOLATION WILL BE USED TO CORRECT THE CHOSEN
C     'BASE' FERTILITY RATES ACCORDING TO ITS POSITION RELATIVE
C     TO THE AVERAGE YEAR.
C     NOW THE ACTUAL DATA MUST BE WRITTEN TO THE OUTPUT FILE
C     TO BE INCORPORATED WITH THE INTERPOLATED VALUES.
      LM = BASE + CYCLE
      MU = (BASEY - LM)/CYCLE + 1
      IF (MU .LT. 1) MU = 1
      DO 70 JJ = 1,MU
      WRITE (6,102) AVCFS(JJ)
102   FORMAT ('  THE CFS IS = ',1X,F5.2)
      DO 72 I = 1,2
      DO 71 J = 1,6
      ASFR(I,J,JJ) = .2 * PROP(I,J) * AVCFS(JJ)
71    IASFR(I,J,JJ) = IFIX(10000.*ASFR(I,J,JJ))
72    IF(BASEY.GT.BASE+CYCLE) WRITE (10,110) (IASFR(I,KK,JJ),KK=1,6)
110   FORMAT (6I4)
85    DO 300 I=1,2
      IF (I.EQ.2 .AND. RACES .EQ.1) GO TO 300
      WRITE (6,950)
950   FORMAT ('1   VALUES COMPUTED USING UPDATED INFORMATION',//)
      IF (I .EQ.1 .AND. RACES .EQ. 2) WRITE (6,960)
960   FORMAT (' WHITE FEMALE',//)
      IF (I .EQ. 2 .AND. RACES .EQ. 2) WRITE (6,970)
970   FORMAT (' NONWHITE FEMALE',//)
      IDTE = (JJ*CYCLE) + BASE
      WRITE (6,975) IDTE
975   FORMAT ('  AGE ',10X,I5,/
     X          16X, 'FERT.',/
     X          16X,'RATES',//)
      A1 = 0.
      DO 280 KK = 1,6
      LL = 10 + 5 * KK
      LLL = LL + 4
      WRITE (6,980) LL, LLL, ASFR(I,KK,JJ)
      A1 = A1 + ASFR (I,KK,JJ)
980   FORMAT (1X,I2,'-',I2,10X,F5.4)
280   CONTINUE
      A1 = A1*5.
      WRITE (6,985) A1
985   FORMAT (//,' CFS',11X,F5.2)
300   CONTINUE
70    CONTINUE
      DO 83 I=1,2
      DO 82 K=1,6
C     THE NEW 'BASE RATES' WILL BE THE LAST TO BE COMPUTED HERE
C     THIS NEW BASE WILL BE USED IN THE INTERPOLATIONS IN MAIN
82    BASEFR(I,K) = ASFR(I,K,MU)
      IF (I .EQ. 2 .AND. RACES .EQ. 1) GO TO 83
      IF (I .EQ.1 .AND. RACES .EQ. 2) WRITE (6,960)
      IF (I .EQ. 2 .AND. RACES .EQ. 2) WRITE (6,970)
      WRITE (6,986) IDTE,(BASEFR(I,J),J=1,6)
83    CONTINUE
986   FORMAT ('0  NEW BASE YEAR FERTILITY RATES',//,10X,I4,
     X10X,'FERTILITY RATE',//,6(28X,F6.4,/))
      IF (BASE .EQ. BASEY) MU = 0
      RETURN
```

```
END
```

SURV: The Survival Rate Generator

Purpose

The generator creates age specific survival rates for each of the cycles in the APPLE projection period. Input consists of run parameters describing numbers of cycles and cycle length, base year mortality rates, and rate change indicators for the end of forecast year.

Procedure

The generator is composed of (1) a main program SURV, (2) subroutine TITLE, and (3) subroutine LIFE.

Program SURV reads input parameters, base year, and percentage change data. In a manner similar to that employed in LFPR and HHEAD, the first computation step is the development of end of rate change forecast period rates, accomplished by applying percent of death rate remaining figures, to base year death rates. Having established base year death rates and end projection period death rates, the subroutine LIFE is called to produce abridged life tables. LIFE (written by Holly McKercall-Hanks) uses the method in An Introduction to Stochastic Processes in Biostatistics by Dr. C.L. Chiang (pages 203-208). Basically the method involves computation of survival rates from mortality rates. Additional statistics computed are: years of life remaining at a particular age, number of years lived in a particular interval, expected life span, and similar measures. Once these survival rates have been computed for base year and end of forecast year, a simple linear interpolation routine in SURV is used to calculate the forecast period cycle rates. As in FERT, LFPR, and HHEAD, the period rates are needed for each mid-cycle therefore survival rates are computed for mid 1972, mid 1977, etc. (or mid 1970, mid 1971, etc. if the cycle length is one year). For any cycles falling outside the forecast period, the survival rates are assumed to remain constant.

SURV

Dictionary of Variables

BASEDR	-Base year death rates an array of dimension (races * sex, 19)
CHANDR	-Percentage of death rate remaining at the end of the death rate change period an array of dimension (races * sex, 19)
BASE	-Base year of APPLE Run
END	-End year of APPLE Run
DRCO	-Base year of death rate change projection
DRC9	-End year of death rate change projection
CYCLE	-Cycle of output, value 1 or 5
RACES	-Number of races (white/non white or total) considered, value 1 or 2
DR9	-End year death rates an array of dimension (sex * races, 19)
AGE	-Age intervals, data input in stream of program
SRATEO(I)	-Survival rate from age I to age I+N (cycle) for the Base Year
CAPL(I)	-Capital L (Chiang reference) number of years lived in interval X (I) to X (I+1) by the survivors at age X. Also gives stationary population values
SRATE9(I)	-Survival rates at end of the death rate change projection
SRATEC	-Survival rates for 5 year cycle
YEAR	-End year of cycle
SRATES	-Survival rates for single years (when Cycle = 1)

FIGURE 5
SURV PROGRAM FLOW

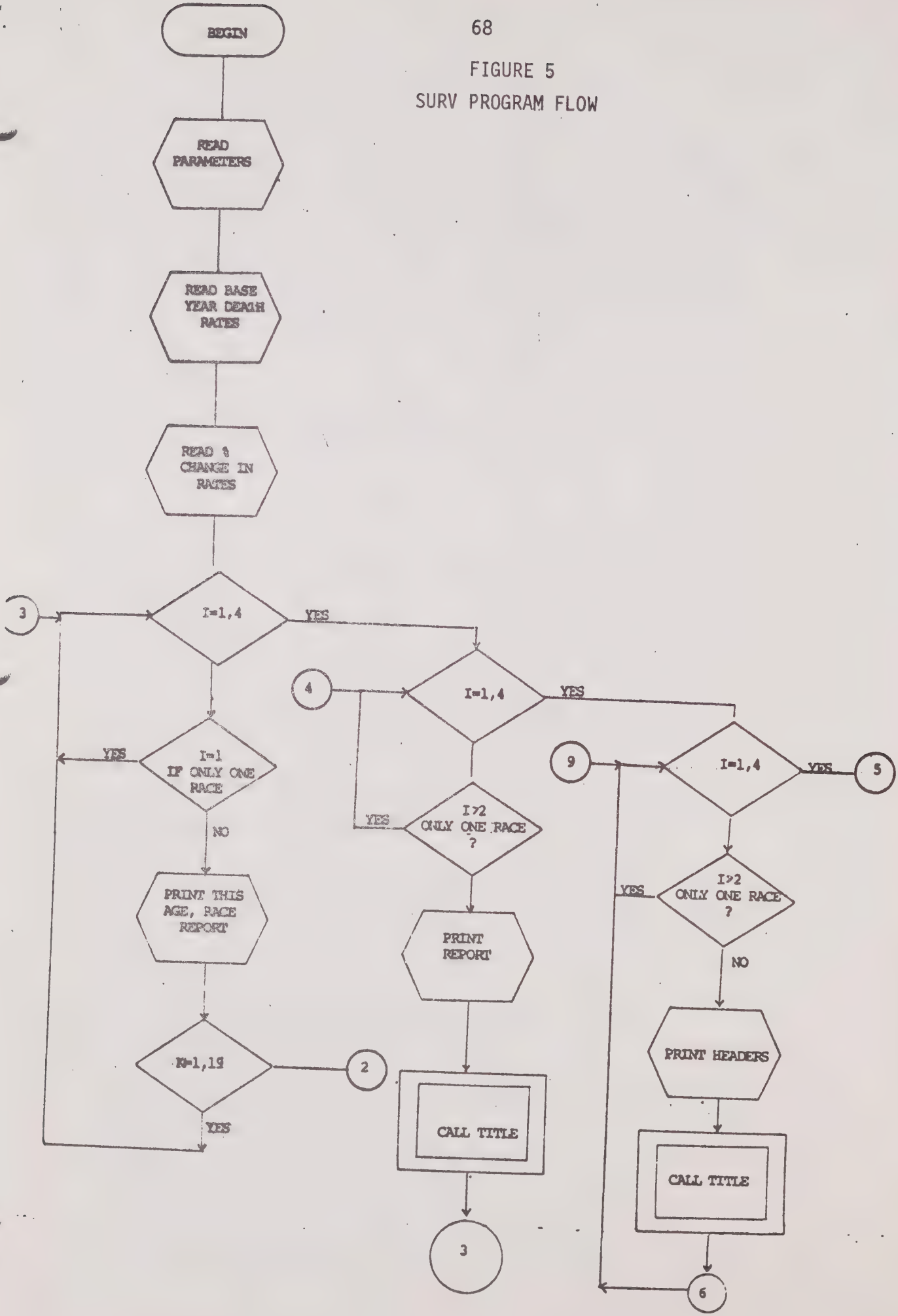


FIGURE 5

(Continued)

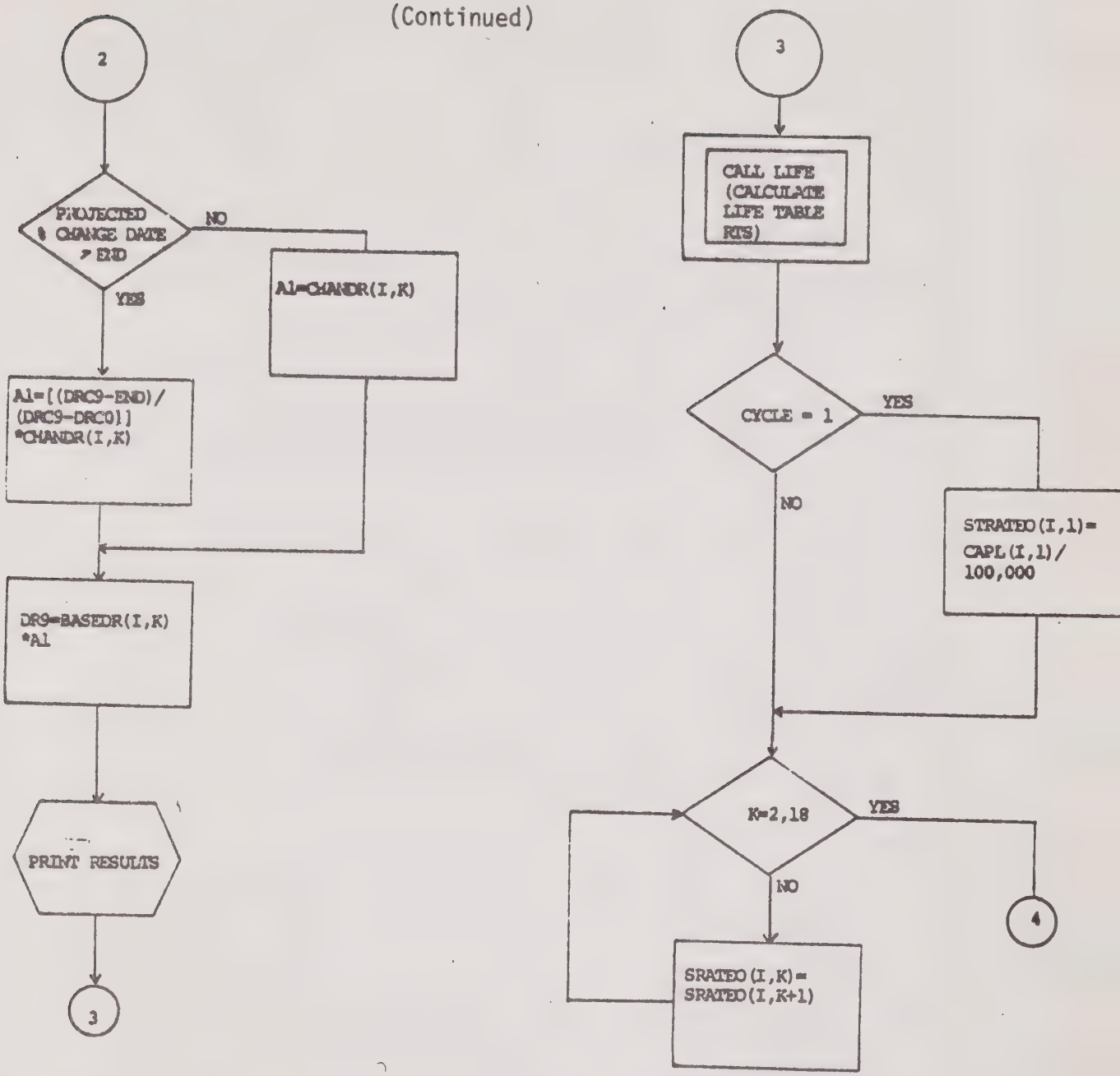


FIGURE 5

(Continued)

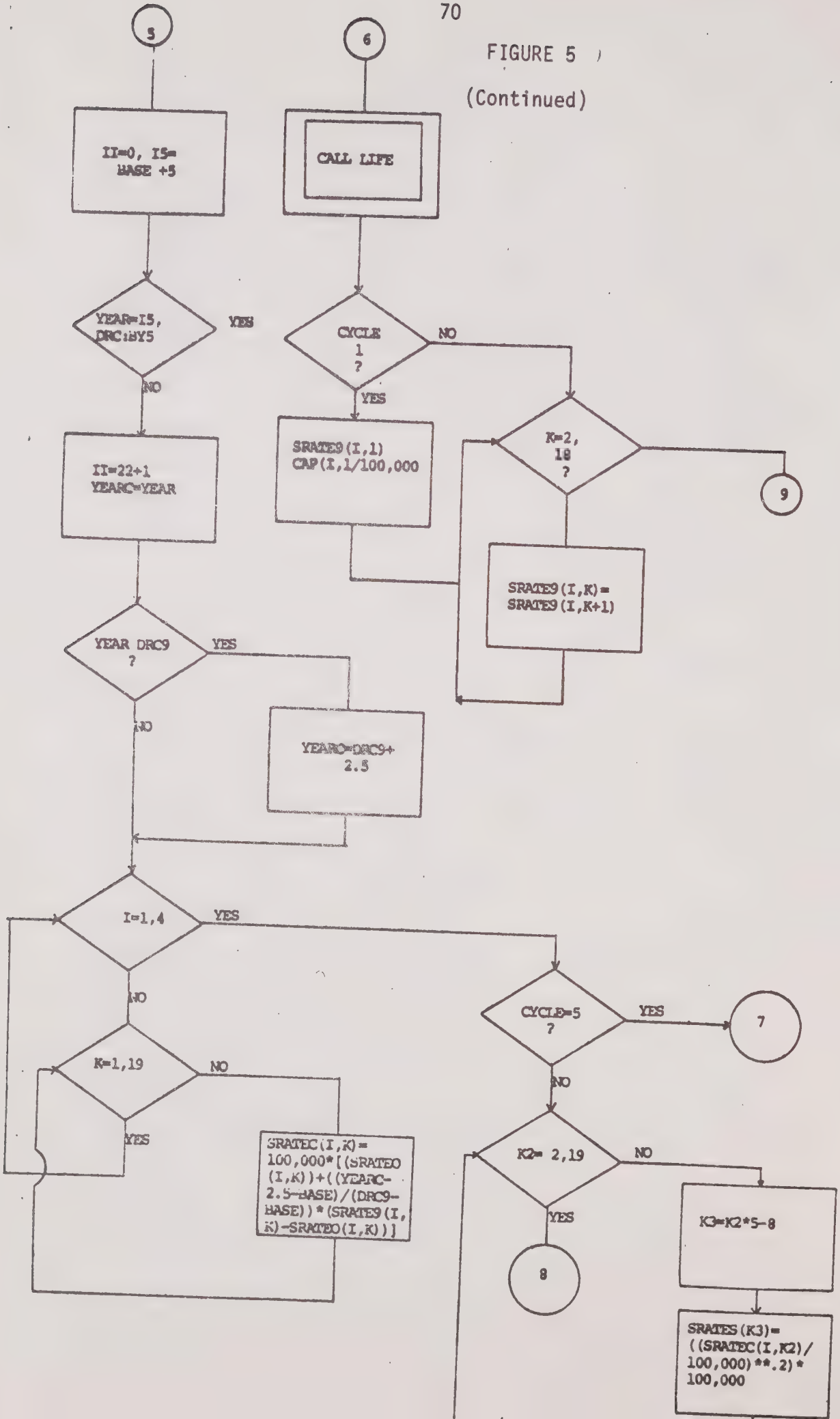
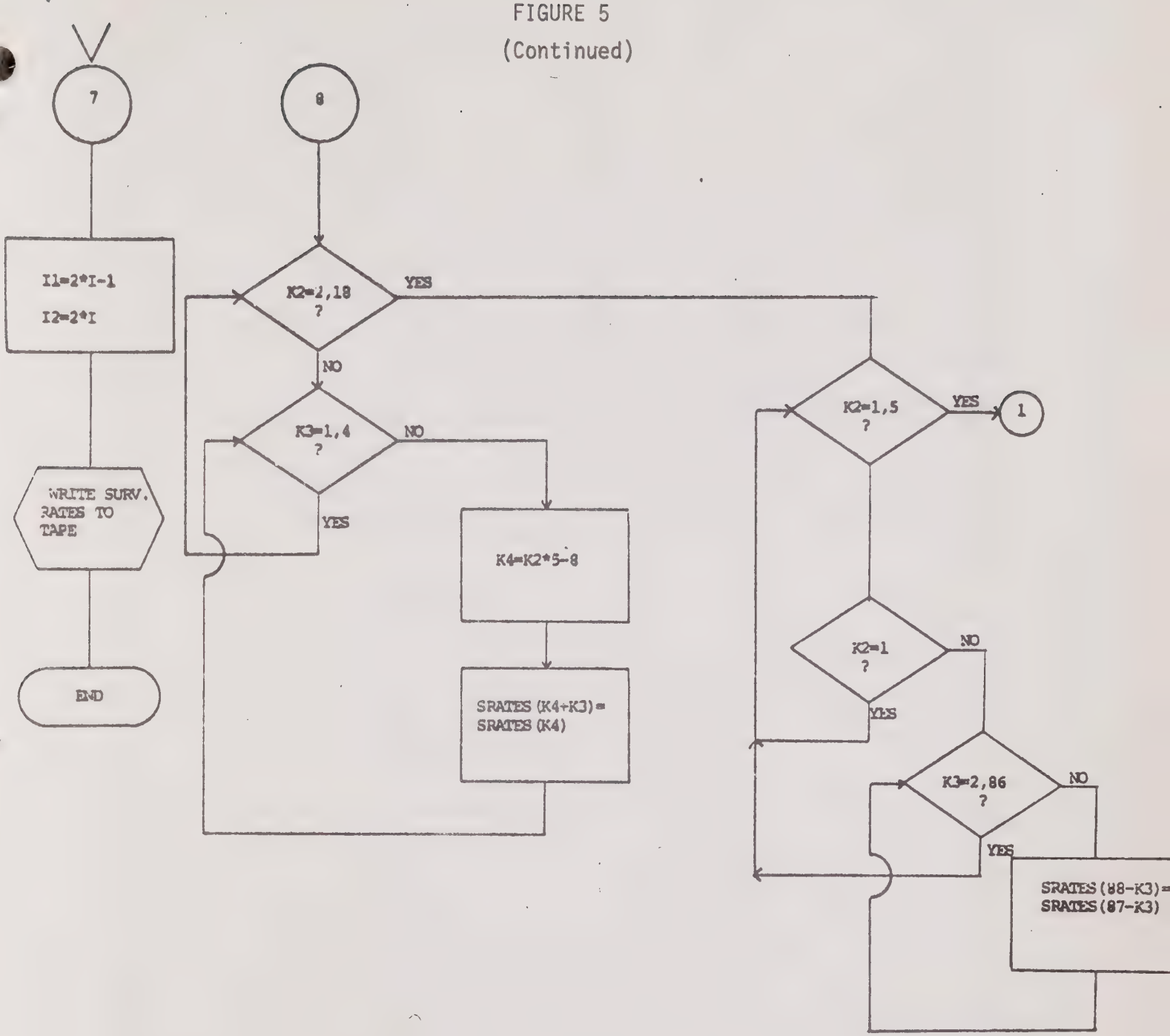


FIGURE 5
(Continued)



SURV Program

Input Description

The first input cards are three data description titles, titling the run, identifying the base year mortality rates, and identifying the projected change in death rates. The parameter record contains information regarding base year, end year of APPLE run, base year of death rate change projected, end year of death rate change projected, number of races, and length of cycle. The data to be used in calculation is read next. This data lists base year death rates and percent of death rate remaining at the end of the death rate change period.

The following outlines input requirements:

CARD	COLUMNS	DESCRIPTION
1		TITLE CARD 1 (20A4)
	1-80	Any title
2		TITLE CARD 2
	1-80	Any title
3		TITLE CARD 3
	1-80	Any title
4		PARAMETER (3I4,2I2,I4)
	1-4	Base year

CARD	COLUMNS	DESCRIPTION
	5-8	End year
	9-12	End year of rate change projected
	13-14	Cycle length
	15-16	Races (1 or 2)
	17-20	Base year of death rate projected
5		DEATH RATES, white males, birth to age 64 (base year deaths per population by age and sex) (12F6.6)
1-6		0-1 death rate, white males
7-12		1-4 death rate, white males
13-18		5-9 death rate, white males
19-24		10-14 death rate, white males
25-30		15-19 death rate, white males
31-36		20-24 death rate, white males
37-42		25-29 death rate, white males
43-48		30-34 death rate, white males
49-54		35-39 death rate, white males
55-60		40-44 death rate, white males
61-66		45-49 death rate, white males
67-72		50-54 death rate, white males
6		DEATH RATES, white males, ages 65-85 and over (base year deaths per population by age and sex) (7F6.6)
1-6		55-59 death rate, white males
7-12		60-64 death rates, white males
13-18		65-69 death rates, white males
19-24		70-74 death rates, white males
25-30		75-79 death rates, white males
31-36		80-84 death rates, white males
37-42		85 and over death rates, white males

CARD	COLUMNS	DESCRIPTION
7		DEATH RATES, white females, birth to age 54
		Same as Card No. 5
8		DEATH RATES, white females, ages 55-85 and over
		Same as Card No. 6
9		DEATH RATES, nonwhite males, birth to age 54
		Same as Card No. 5
10		DEATH RATES, nonwhite males, ages 55-85 and over
		Same as Card No. 6
11		DEATH RATES, nonwhite females, birth to age 54
		Same as Card No. 5
12		DEATH RATES, nonwhite females, ages 55-85 and over
		Same as Card No. 6

6 - proportion of base year (base year of "death rate change" projection) death rate remaining at end year of "death rate change" projection.

CARD	COLUMNS	DESCRIPTION
13		PROPORTION OF DEATH RATE REMAINING AT END OF PROJECTION, white males, birth to age 85+ (19F3.3)
1-3		0-1 proportion, white males
4-6		1-4 proportion, white males
7-9		5-9 proportion, white males
10-12		10-14 proportion, white males
13-15		15-19 proportion, white males
16-18		20-24 proportion, white males
19-21		25-29 proportion, white males
22-24		30-34 proportion, white males
25-27		35-39 proportion, white males
28-30		40-44 proportion, white males

CARD	COLUMNS	DESCRIPTION
31-33		45-49 proportion, white males
34-36		50-54 proportion, white males
37-39		55-59 proportion, white males
40-42		60-64 proportion, white males
43-45		65-69 proportion, white males
46-48		70-74 proportion, white males
49-51		75-79 proportion, white males
52-54		80-84 proportion, white males
55-57		85 and over proportion, white males
14		PROPORTION OF DEATH RATE REMAINING AT END OF PROJECTION, WHITE FEMALES, BIRTH TO AGE 85+
		Same as Card No. 13
15		PROPORTION OF DEATH RATE REMAINING AT END OF PROJECTION, nonwhite males, birth to age 85+
		Same as Card No. 13
16		PROPORTION OF DEATH RATE REMAINING AT END OF PROJECTION, nonwhite females, birth to age 85+
		Same as Card No. 13

Output Description

SURV generates two output files, a print file and output file number 13. The report printed lists run parameters, base year mortality rates, computed base year survival rates, end year mortality rates, end year survival rates, and percentage of death rate remaining at the end of the death rate change projection period. File 13 contains survival rates written to five significant digits. These records serve as input to the cohort survival model. The format of the output file is similar in field length and record length to the input records containing base year mortality rate information.

Technical Specifications

SURV reads file 5, SYSIN, and writes the report to file 6, SYSOUT. The output data file is written to file 13. Below is the JCL (Job Control Language) necessary to generate these files when executing SURV:

```
//FT05F001  DD  DDNAME=SYSIN
//FT06F001  DD  SYSOUT=A
//FT13F001  DD  DSN=name,UNIT=unit type, SPACE=(TRK,(5,1),RLSE),
//          VOL=SER=Volname,DISP=(,CATLG),DCB=(LRECL=80,BLKSIZE=3520,
//          RECFM=FB)
//SYSIN     DD *
  data
/*
```

Timing and core requirements for execution of SURV are:

CPU TIME	.021
REGION USED	38K

```
C
C     PROGRAM TO GENERATE SURVIVAL RATES BASED
C     ON INPUT OF BASE YEAR DEATH RATES AND
C     FORECAST OF PROPORTION OF DEATH RATE
C     REMAINING AT A FUTURE DATE
C
      REAL REM(20),BASEDR(4,19),CHANDR(4,19),BASDR(19),DR9T(19)
      REAL DR9(4,19),AGE(38),SRATE0(4,19), CAPL(4,19)
      REAL SRATE9(4,19),SRAT0(19),SRAT9(19),CAPLT(19)
      INTEGER BASE, END, DRC0, DRC9, CYCLE, RACES, YEAR
      INTEGER SRATEC (4,19), SRATES(87)
      DATA SRATE0/76*0.0/
      DATA SRATE9/76*0.0/
C     READ IN AGE CATEGORY LABELS
      DATA AGE/3H  0,3H-1 ,3H  1,3H-4 ,3H  5,3H-9 ,3H 10,3H-14,
     1         3H 15,3H-19,3H 20,3H-24,3H 25,3H-29,3H 30,3H-34,
     2         3H 35,3H-39,3H 40,3H-44,3H 45,3H-49,3H 50,3H-54,
     3         3H 55,3H-59,3H 60,3H-64,3H 65,3H-69,3H 70,3H-74,
     4         3H 75,3H-79,3H 80,3H-84,3H 85,3H-+  /
C     BASE          -    BASE YEAR OF APPLE RUN
C     END           -    ENDYEAR OF APPLE RUN
C     DRC0          -    BASE YEAR OF DEATH RATE CHANGE FORECAST
C     DRC9          -    ENDYEAR OF DEATH RATE CHANGE FORECAST
C     CYCLE         -    "1" OR "5" YEAR CYCLES OF OUTPUT
C     RACES         -    "1" OR "2" RACES CALCULATED
C     BASEDR        -    APPLE BASE YEAR DEATH RATES
C     DRC           -    ABBREVIATION FOR DEATH RATE CHANGE
C     CHANDR        -    PROPORTION CHANGE IN DEATH RATE
C     DR9           -    DRC ENDYEAR DEATH RATES
C     AGE           -    AGE INTERVALS
C     SRATE0(I)     -    SURVIVAL RATE FROM AGE I TO AGE I PLUS N(BASE
C                        YEAR
C     CAPL(I)       -    CAPITOL L, STATIONATY POPULATION AT AGE I
C     SRATE9(I)     -    SURVIVAL RATES AT ENDOF DRC FORECAST
C     SRATEC        -    SURVIVAL RATES FOR 5 YEAR CYCLE
C     YEAR          -    ENDYEAR OF CYCLE
C     SRATES        -    SURVIVAL RATES FOR SINGLE YEARS
C     READ RUN DESCRIPTION
      READ (5,900) REM
      WRITE (6,901) REM
900   FORMAT (20A4)
901   FORMAT ('1','SURVIVAL RATE GENERATOR FOR APPLE',//,1X,
     * 20A4,/)
C     READ DESCRIPTION OF BASE YEAR DEATH RATES
      READ (5,900) REM
      WRITE (6,902) REM
902   FORMAT (' ',20A4,/)
C     READ DESCRIPTION OF CHANGE IN DEATH RATES
      READ (5,900) REM
      WRITE (6,902) REM
C     READ PARAMETER CARD
      READ (5,903) BASE, END, DRC9, CYCLE,  RACES, DRC0
903   FORMAT (3I4,2I2,I4)
      WRITE (6,904) BASE, END, DRC0, DRC9, CYCLE, RACES
```

```
  904   FORMAT (' RUN PARAMETERS',/,' APPLE BASE',
       X' YEAR-',I6,/,' APPLE ENDYEAR-', I6,/,
       X' DRC BASE YEAR-', I6,/,' DRC ENDYEAR',I6,/,
       X' YEARS PER CYCLE-',I3,/,' RACES-',I3)
C      READ APPLE BASE YEAR DEATH RATES
       READ (5,905)  ((BASEDR(I,J),J=1,19),I=1,4)
  905  FORMAT (12F6.6,/,7F6.6)
C      READ PROPORTION OF CHANGE INDEATH RATES
C      BETWEEN BASE YEAR OF DRC FORECAST AND END
C      YEAR OF DRC FORECAST
       READ (5,908) ((CHANDR(I,J),J=1,19),I=1,4)
  908  FORMAT (19F3.3)
C      APPLY PROPORTIONAL CHANGE WHICH OCCURS DURING FORECAST TO APPLE
C      BASE YEAR DEATH RATES TO DERIVE DRC ENDYEAR DEATH RATES.
C      CALCULATE DEATH RATES AT ENDOF DRC FORECAST YEAR
       DO 100  I=1,4
       IF (RACES .EQ. 1 .AND. I .GT. 2) GO TO 100
       WRITE(6,909)
  909  FORMAT(' INPUT AND CALCULATED VALUES',//)
       WRITE(6,910) BASE,DRC0,DRC9,BASE,DRC9,DRC9
  910  FORMAT('  AGE  ',10X,I5,10X,I5,'-',I4,10X,I5,'-',I4,10X,I5,/,
      1 15X,'DEATH',10X,'PROPORTION',10X,'PROPORTION',10X,'DEATH',/,
      2 15X,'RATES',10X,'  CHANGE  ',10X,'  CHANGE  ',10X,'RATES',//)
       CALL TITLE (I,RACES)
       DO 90  K=1,19
       IF (DRC9-END) 701,701,700
  700  A1 = ((DRC9-END)/(DRC9-DRC0)) * CHANDR(I,K)
       DR9(I,K) = BASEDR(I,K) * A1
REPLACE
       WRITE (6,915) AGE(2*K-1),AGE(2*K),BASEDR(I,K),CHANDR(I,K),A1,
      1 DR9(I,K)
       GO TO 705
  701  DR9(I,K) = BASEDR(I,K)  * CHANDR(I,K)
C      PRINT OUT INPUT AND CALCULATED VALUES
       IF (RACES .EQ. 1 .AND. I .GT. 2) GO TO 90
       WRITE (6,915) AGE(2*K-1),AGE(2*K),BASEDR(I,K),CHANDR(I,K),
      1 CHANDR(I,K), DR9(I,K)
  705  CONTINUE
  915  FORMAT(1X,2A3,7X,4(F10.5,10X))
  90   CONTINUE
  100  CONTINUE
       DO 200 I=1,4
       IF(RACES .EQ. 1 .AND. I .GT. 2) GO TO 200
       WRITE (6,920) BASE
  920  FORMAT ('1LIFE TABLE FOR BASE YEAR',I5,/)
       CALL TITLE (I,RACES)
C      CALCULATE LIFE TABLE VALUES FOR APPLE BASE YEAR
C      THIS NEXT STEP STRIPS ONE ROW AT A TIME
       DO 222 J = 1,19
       BASDR(J)=BASEDR(I,J)
  222  CONTINUE
       CALL LIFE(BASDR,SRATO,CAPLT)
       DO 444 J=1,19
       SRATEO(I,J) = SRATO(J)
       CAPL(I,J) = CAPLT(J)
  444  CONTINUE
```

```
      IF (CYCLE .EQ. 1) SRATE0(I,1) = CAPL(I,1) / 100000.
C     ELIMINATE SRATE(2),   MAKE SRATE(18) AND SRATE(19) THE SAME
      DO 150 K= 2,18
      SRATE0(I,K) = SRATE0(I,K+1)
150   CONTINUE
200   CONTINUE
C     CALCULATE LIFE TABLE VALUES FOR DRC ENDYEAR OF FORECAST
      DO 300 I=1,4
      IF(RACES .EQ. 1 .AND. I .GT. 2) GO TO 300
      WRITE (6,930) DRC9
930   FORMAT('1LIFE TABLE FOR END OF DEATH RATE CHANGE FORECAST YEAR',
     X I5,/)
      CALL TITLE (I,RACES)
C     THIS NEXT STEP STRIPS ONE ROW AT A TIME
      DO 333 J=1,19
      DR9T(J) = DR9(I,J)
333   CONTINUE
      CALL LIFE(DR9T,SRAT9,CAPLT)
      DO 555 J = 1,19
      SRATE9(I,J) = SRAT9(J)
      CAPL(I,J) = CAPLT(J)
555   CONTINUE
      IF (CYCLE .EQ. 1) SRATE9(I,1) = CAPL(I,1) / 100000.
C     ELIMINATE SRATE(2),   MAKE SRATE(18) AND SRATE(19) THE SAME
      DO 250 K=2,18
      SRATE9(I,K) = SRATE9(I,K+1)
250   CONTINUE
300   CONTINUE
C     PRODUCE MID INTERVAL SURVIVAL RATES FOR EACH 5 YEAR CYCLE
C     USES LINEAR INTERPOLATION
      II = 0
      I5 = BASE + 5
      DO 400 YEAR = I5,DRC9,5
      II = 22 + 1
      YEARC = YEAR
      IF(YEAR .GT. DRC9) YEARC = DRC9 + 2.5
      DO 390 I=1,4
      DO 340 K=1,19
      SRATEC(I,K) = 100000.*(SRATE0(I,K)+((YEARC-2.5-BASE)/
     1 (DRC9-BASE))*(SRATE9(I,K) - SRATE0(I,K)))  + .5
340   CONTINUE
      IF (CYCLE .EQ. 5) GO TO 385
C     PRODUCE SINGLE YEAR RATES FROM MID INTERVAL RATES
      SRATES(1) = ((SRATEC(I,1)/100000.)**.2) *100000.  +.5
      DO 350 K2 = 2,19
      K3 = K2 * 5 -8
      SRATES(K3) = ((SRATEC(I,K2) / 100000.)**.2) * 100000.  +.5
350   CONTINUE
      DO 360 K2 = 2,18
      DO 360 K3 = 1,4
      K4 = K2*5 -8
      SRATES(K4 + K3) = SRATES(K4)
360   CONTINUE
      DO 370 K2 = 1,5
      IF(K2 .EQ. 1) GO TO 366
      DO 365 K3 = 2,86
```

```
      SRATES(88-K3) = SRATES(87-K3)
365   CONTINUE
      SRATES(1) = ((SRATEC(I,1)/100000.)**(1./(6.-K2)))*100000. +.5
366   CONTINUE
      I1 = 20 * I - 10
      WRITE (13,980) (SRATES(L),L= 1,14),II,I1
980   FORMAT (14F7.5,5X,I2,I3)
      I1 = I1 + 1
      WRITE (13,980) (SRATES(L),L=15,28),II,I1
      I1 = I1 + 1
      WRITE (13,980) (SRATES(L),L=29,42),II,I1
      I1 = I1 + 1
      WRITE (13,980) (SRATES(L),L=43,56),II,I1
      I1 = I1 + 1
      WRITE (13,985) (SRATES(L),L=57,66),II,I1
985   FORMAT (10F7.5,25X,I2,I3)
      I1 = 20* I
      WRITE (13,980) (SRATES(L),L=67,80),II,I1
      I1 = I1 + 1
      WRITE (13,988) (SRATES(L),L=81,87),II,I1
988   FORMAT (7F7.5,40X,I2,I3)
370   CONTINUE
385   I1 = 2*I-1
      I2 = 2*I
      WRITE (13,950) (SRATEC(I,L),L=1,14),II,I1,
     1  (SRATEC(I,LL),LL=15,19),II,I2
950   FORMAT (14I5,5X,I2,I2,1X,/,5I5,50X,2I2,1X)
390   CONTINUE
400   CONTINUE
      ENDFILE 13
      REWIND 13
      STOP
      END
```

```
      SUBROUTINE TITLE (I,IRACES)
C     PRINT SEX AND RACE ON REPORT
901   FORMAT(' WHITE MALES',//)
902   FORMAT(' WHITE FEMALES',//)
903   FORMAT(' NONWHITE MALES',//)
904   FORMAT(' NONWHITE FEMALES',//)
905   FORMAT(' MALES',//)
906   FORMAT(' FEMALES',//)
      IF(IRACES .EQ. 1) GO TO 100
      IF(I .EQ. 1) WRITE(6,901)
      IF(I .EQ. 2) WRITE(6,902)
      IF(I .EQ. 3) WRITE(6,903)
      IF(I .EQ. 4) WRITE(6,904)
      RETURN
100   CONTINUE
      IF(I .EQ. 1) WRITE(6,905)
      IF(I .EQ. 2) WRITE(6,906)
      RETURN
      END
```

```
      SUBROUTINE LIFE(M,SRATE1,CAPL)
C
C     PROGRAM TO CONSTRUCT AN ABRIDGED LIFE TABLE USING THE METHOD
C     SUGGESTED IN (AN INTRODUCTION TO STOCHASTIC PROCESSES IN
C     BIOSTATISTICS( BY  DR. C.L. CHIANG, CHAPTER 9, PAGES 203 -208
C
C
C
C     WRITTEN BY HOLLY HOLLINGSHEAD , APRIL,1971 FOR SANTA CLARA COUNTY
C     PLANNING DEPARTMENT, RESEARCH SECTION
C
C
C
      REAL QHAT(19),D(19),N(18),M(19),A(18),CAPL(19),L(19),E(19)
      REAL T(19),SRATE1(19)
      INTEGER LL(19),DD(19),CAPLL(19),TT(19)
C
C
C
      DIMENSION AGE(38)
13    FORMAT (5X,2A3,5X,F9.7,5X,I6,5X,I5,5X,F4.2,5X,I6,5X,I7,5X,
     X    F5.2,5X,F8.6,5X,F8.6)
14    FORMAT (5X, 2A3, 5X, F9.7,5X, I6, 5X, I5, 5X,       9X,I6,5X,I7,5X
     X    F5.2,5X,F8.6,5X,F8.6)
16    FORMAT (6X,'AGE',8X,'% DYING',6X,'NUMBER',4X,'NUMBER',
     X3X,'INTERVAL',3X,'NUMBER',4X,'YEARS OF',4X,'YEARS',5X,
     X'SURVIVAL',/,18X,'DURING',4X,'SURVIVING',2X,'DYING IN',
     X4X,'% OF',4X,'OF YEARS',3X,'ALL LIVES',3X,'OF LIFE',6X,
     X'RATE',/,17X,'INTERVAL',4X,'INTERVAL',3X,'INTERVAL',3X,
     X'LIFE',5X,'LIVED',4X,'REMAINING',2X,'REMAINING')
C
C     READ IN N(I), LENGTH OF THE I-TH INTERVAL
      DATA N/1,4,16*5/
C     READ IN A(I),FRACTION OF LAST AGE INTERVAL OF LIFE
      DATA A/.10,.39,.46,.54,.57,.49,.50,.52,.54,.54,.54,.53,.52,
     1       .52,.52,.51,.51,.45/
C     READ IN AGE CATEGORY LABELS
      DATA AGE/3H  0,3H-1 ,3H  1,3H-4 ,3H  5,3H-9 ,3H 10,3H-14,
     1         3H 15,3H-19,3H 20,3H-24,3H 25,3H-29,3H 30,3H-34,
     2         3H 35,3H-39,3H 40,3H-44,3H 45,3H-49,3H 50,3H-54,
     3         3H 55,3H-59,3H 60,3H-64,3H 65,3H-69,3H 70,3H-74,
     4         3H 75,3H-79,3H 80,3H-84,3H 85,3H-+  /
C
      DO 99 I= 1,19
      E(I) = 0.
      T(I) = 0.
99    CONTINUE
C
C     CALCULATE QHAT(I),PROPORTION DYING DURING INTERVAL . FOR
C     DERIVATIONS OF FORMULAE ,PLEASE SEE ABOVE -CITED REFERENCE.
C
C
C
      DO 50 I=1,18
```

```
      QHAT(I) = (N(I)* M(I))/(1.0 + (1.0 - A(I)) * N(I)* M(I))
   50 CONTINUE
      QHAT(19) = 1.0
C     PROPORTION DYING  IN 85+ AGE GROUP IS UNITY.
C
C     USING  THE QHAT(I), CALCULATE THE D(I),NUMBER OF DEATHS IN INTERV
C     X(I) TO  X(I+1), AND (L(I), NUMBER  OF PERSONS SURVIVING TO EXACT
C     AGE  MARKING BEGINNING OF  EACH AGE INTERVAL.
C
C     DUE TO THE PECULIARITIES OF FORTRAN, IT IS NOT POSSIBLE TO USE ZE
C     FOR AN ARRAY SUBSCRIPT. THUS, THE FIRST VALUES IN EACH COLUMN OF
C     THESE LIFE TABLES WHICH WOULD NORMALLY BE SUBSCRIPTED WITH A ZERO
C     (E.G. L(0) ) WILL BE;REFERRED TO WITH SUBSCRIPT OF 1.
C
C
      L(1) = 100000.0
      D(1) = QHAT (1) * L(1)
      DO 60  I= 1,18
      L(I+1) = L(I) - D(I)
      D(I+1) = L(I+1) * QHAT (I+1)
   60 CONTINUE
C
C
C     CALCULATE CAPL(I),NUMBER OF YEARS LIVED IN INTERVAL X(I) TO X(I+1
C     BY THE L(I) SURVIVORS AT AGE  X(I)| CAPL (I) ALSO GIVES STATIONAR
C     POPULATION VALUES.
C
      DO 65  I= 1,18
      CAPL(I) =  N(I) * ( L(I)-D(I)) + A(I) * N(I) * D(I)
   65 CONTINUE
      CAPL(19) = L(19)/M(19)
C
C
C     CALCULATE T(I), TOTAL NUMBER OF YEARS REMAINING TO ALL PEOPLE
C     ATTAINING AGE X(I)
C
      DO 70 I=1,19
      DO 68 J=I,19
      T(I)  =  T(I) +CAPL(J)
   68 CONTINUE
   70 CONTINUE
C
C
C     CALCULATE  E(I),EXPECTATION OF LIFE (AVERAGE  YEARS OF LIFE REMAI
C
      DO 75 I=1,19
      E(I) = T(I) / L(I)
   75 CONTINUE
C
C
C     CALCULATE  S-RATES (PROBABILITY OF SURVIVING  FROM ONE AGE CATEGO
C     TO THE NEXT).
C
C
C
C     USING METHOD IN PREVIOUS LIFE TABLE COMPUTATIONS FOR PLANNING DEP
```

```
C
      SRATE1 (1) =  CAPL(1)/100000.0
      SRATE1 (2) =  CAPL(2)/(CAPL(1)*4.)
      SRATE1 (3) =  CAPL(3) / (CAPL(1)+CAPL(2))
      SRATE1 (19) =  CAPL(19) / (CAPL (18) + CAPL(19))
      DO 85  I= 4,18
      SRATE1(I) = CAPL (I) / CAPL(I-1)
   85 CONTINUE
C
C
C     PRINT LIFE  TABLES
C
C
C     CONVERT CERTAIN VALUES TO INTEGER FORM FOR PRINTING
      DO 90 I=1,19
      LL(I) =      (L(I)+.5)
      CAPLL(I) =       (CAPL(I) +.5)
      DD(I) =       (D(I) +.5)
      TT(I) =       (T(I) +.5)
   90 CONTINUE
C
C     PRINT VALUES
C     PRINT HEADERS
      WRITE (6,16)
      DO  100  I =1,18
      WRITE (6,13) AGE(2*I-1),AGE(2*I),QHAT(I),LL(I),DD(I),A(I),CAPLL(I
     1,TT(I),E(I),SRATE1(I)
  100 CONTINUE
      WRITE (6,14)AGE(37),AGE(38),QHAT(19),LL(19),DD(19),CAPLL(19),TT(1
     1),E(19),SRATE1(19)
C
C
C
C
      RETURN
      END
```

HHEAD: The Household Headship Rate Generator

Purpose

HHEAD generates household headship rates by age and sex specific groups for the APPLE projection period. Base year household headship rates act as input, along with projection and cycle length parameters and exptected change for each age/sex group by the end of the projection period. The generator produces a report of base year rates, final year rates, and the percentage change factors that were input to the program. A file of household headship rates is generated for each cycle of the APPLE projection period.

PPOP -array 4 by 10 containing the percentage change in household headship rates from BASEYR to HHYR

HHINIT -array 4 by 10 containing the base year household headship rates

HH9 -array 4 by 10 containing the computed end year of rate change projection period household headship rates

RACES -number of races (1 indicates no racial breakdown, 2 indicates a white/nonwhite breakdown)

Procedure

HHEAD reads base year data and percentage change data, and then computes end of projection period household headship rates. Mid year rates are developed using linear interpolation. If the end year of the rate change projection period precedes the end year of the APPLE projection period, the remaining cycles are assigned end of projection period values. All of these computed rates are written to file 14 and a user's report is generated.

Dictionary of Variables

BASEYR -base year of APPLE run

GNDYR -end year of APPLE run

HHYR -end year of household headship rate change projection

CYCLE -length of cycle, 1 or 5 years

FIGURE 6

HHEAD PROGRAM FLOW

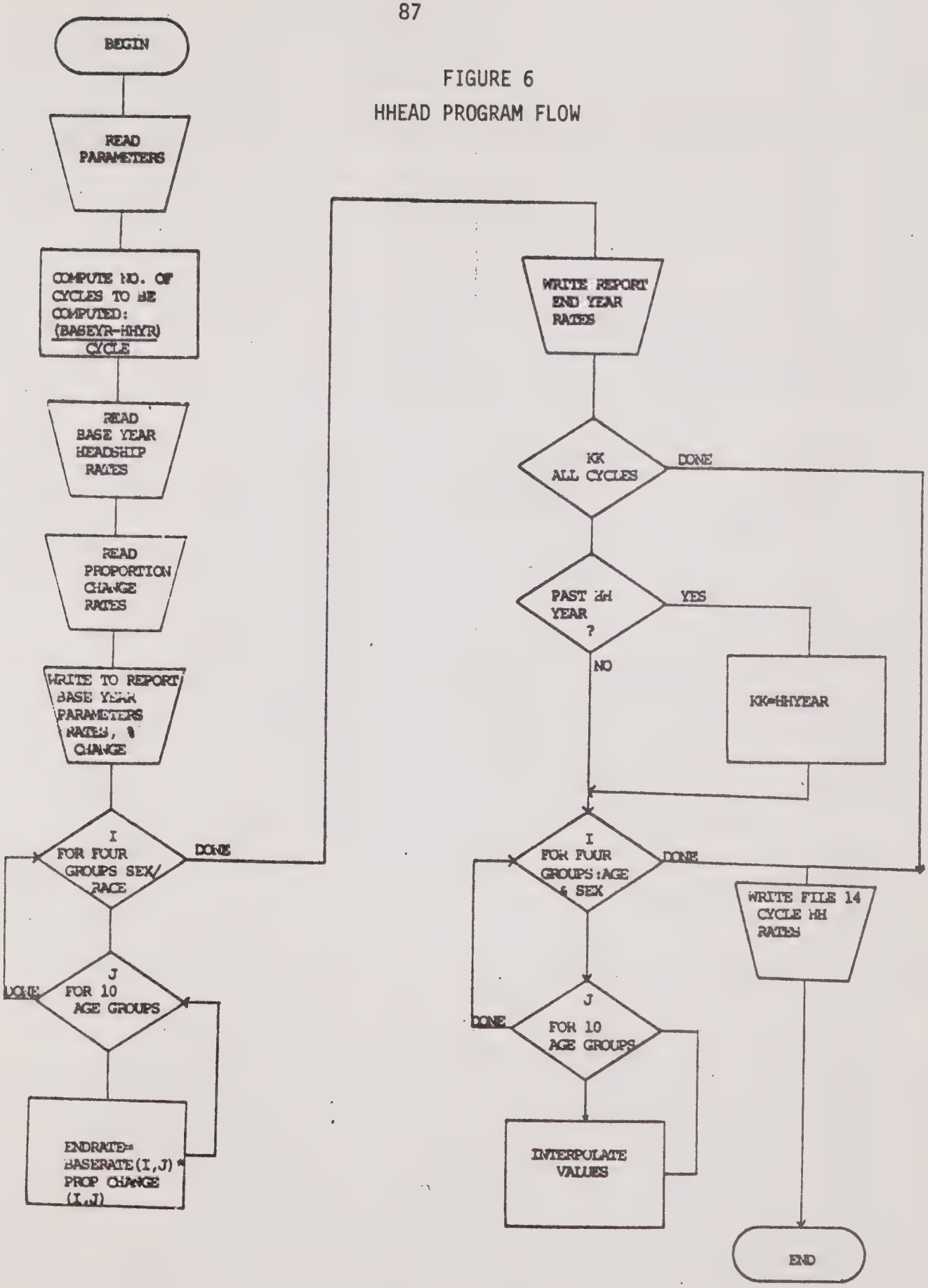
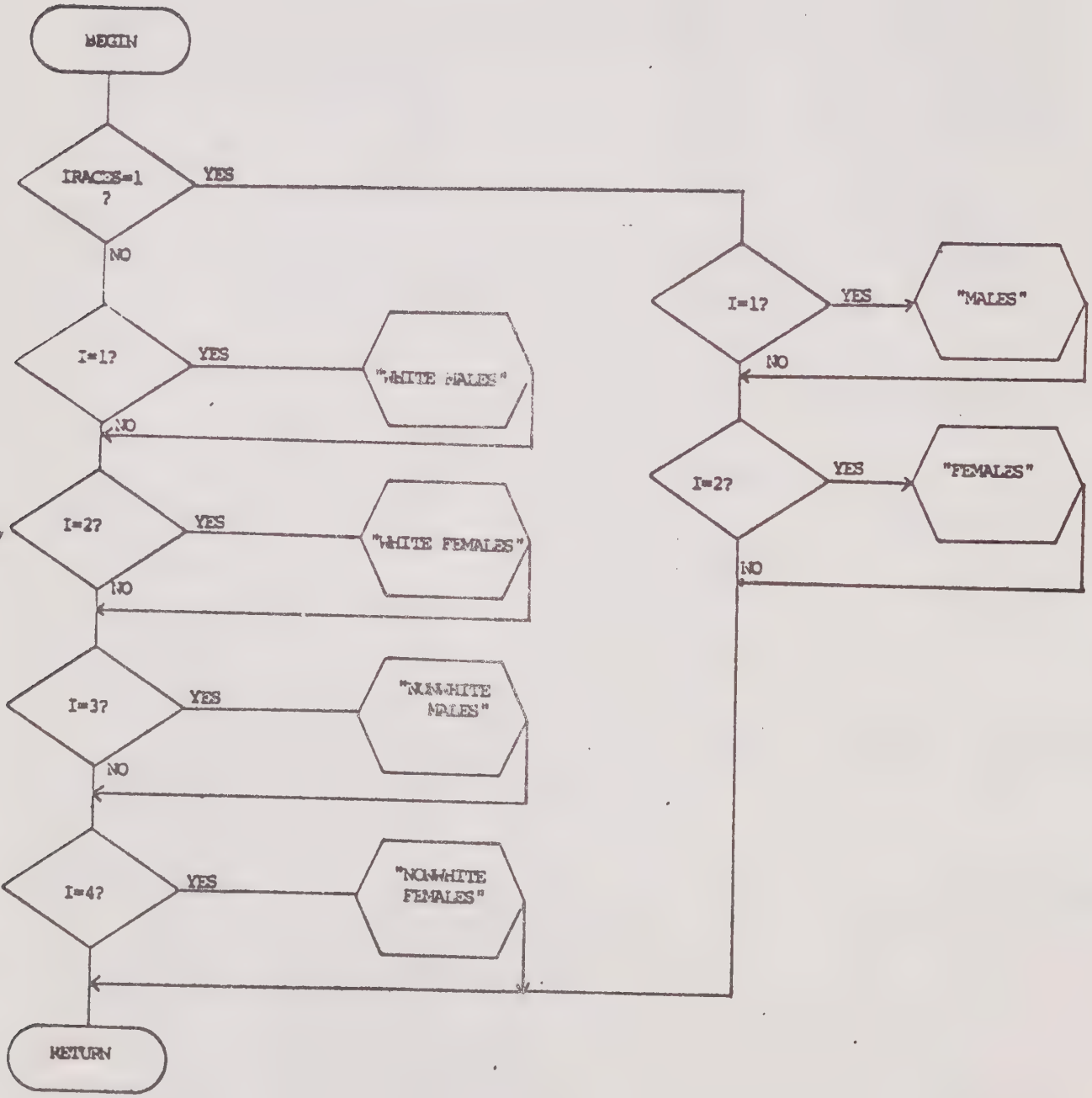


FIGURE 6

(Continued)

TITLE

This is a printing routine that uses RACES as input to determine the extent of reports necessary. Variable RACES is passed to IRACES in this subroutine.



HHEAD Program

Input Description

The input file has three components: (1) parameters, (2) base year rates, and (3) percentage change from base year to end year of forecast.

The parameters indicate the base year for projection, the end year of projection, the end year of household headship rate change, the cycle length, and number of races.

Following is a record format for the input parameters:

CARD	COLUMNS	DESCRIPTION
1		PARAMETERS (3I4, 2I2)
	1-4	Base year for projection
	5-8	End year for APPLE projection
	9-12	End year for Household headship rate change
	13-14	Cycle length (1 or 5)
	15-16	Number of races (1 or 2)

Base year data is input for all age groups, all races and both sexes. (When RACES = 1, records 4, 5, 8, and 9 described below are still required and are blank cards).

CARD	COLUMNS	DESCRIPTION
2		HOUSEHOLD HEADSHIP RATES, WHITE MALES (10I3)
	1-3	0-14 household headship rate, white male
	4-6	15-19 household headship rate, white male
	7-9	20-24 household headship rate, white male
	10-12	25-29 household headship rate, white male
	13-15	30-34 household headship rate, white male
	16-18	35-44 household headship rate, white male
	19-21	45-54 household headship rate, white male
	22-24	55-64 household headship rate, white male
	25-27	65-74 household headship rate, white male
	28-30	75 and over household headship rate, white male
3		HOUSEHOLD HEADSHIP RATES, WHITE FEMALES
		Same as Card No. 2
4		HOUSEHOLD HEADSHIP RATES, NONWHITE MALES
		Same as Card No. 2
5		HOUSEHOLD HEADSHIP RATES, NONWHITE FEMALES
		Same as Card No. 2

Percentage change data is input in a similar manner. The data record is format for this variable.

CARD	COLUMNS	DESCRIPTION
6		CHANGE RATES, white males (10F4.3)
	1-4	0-14 percentage change, white males
	5-8	15-19 percentage change, white males
	9-12	20-24 percentage change, white males
	13-16	25-29 percentage change, white males
	17-20	30-34 percentage change, white males
	21-24	35-44 percentage change, white males
	25-28	45-54 percentage change, white males
	29-32	55-64 percentage change, white males
	33-36	65-74 percentage change, white males
	37-40	75 and over percentage change, white males
7		CHANGE RATES, white females
		Same as Card No. 6
8		CHANGE RATES, nonwhite males
		Same as Card No. 6
9		CHANGE RATES, nonwhite females
		Same as Card No. 6

Output Description

HHEAD produces a report showing the projected change by age, sex, race groups.

Output file number 14 is described in detail below. Each cycle is represented by four file items.

RECORD	FORMAT (10F5.3)
1	Household headship rates for base year + cycle years, white males
2	Same as above, white females
3	Same as above, nonwhite males
4	Same as above, nonwhite females
5-8	Same as above, but for base year + 2 (CYCLE) years
9-12	Same as above, but for base year + 3 (CYCLE) years
13-16	Same as above, but for base year + 4 (CYCLE) years
17-20	Same as above, but for base year + 5 (CYCLE) years
21-24	Same as above, but for base year + 6 (CYCLE) years
(etc. until end year is reached)	

Technical Specifications

HHEAD reads file 5, SYSIN, and writes the report to file 6, SYSOUT. The output data file is written to file 14. Below is the format of the JCL (Job Control Language) necessary to generate these files when executing HHEAD:

```
//FT05F001  DD DDNAME=SYSIN
//FT06F001  DD SYSOUT=A
//FT14F001  DD DSN=name,UNIT=unit type,SPACE=(TRK,(2,1),RLSE),
//          VOL=SER=volname,DISP=(,CATLG),DCB=(LRECL=80,BLKSIZE=3520,
//          RECFM=FB)
//SYSIN     DD *
  data
/*
```

Timing and core requirements for execution of HHEAD are:

CPU TIME	0.014
REGION USED	38K

```
C
C
C     THIS PROGRAM GENERATES HOUSEHOLD HEADSHIP RATES
C     WRITTEN BY NANCY SHYER     AUGUST, 1975
C     INPUT ARE THE BASE YEAR RATES AND THE END YEAR RATES,
C     AS WELL AS THE LENGTH OF CYCLE, END OF FORECAST
C     PERIOD, AND END OF RUN.
C
C     BASEYR    -   BASE YEAR OF APPLE RUN
C     ENDYR     -   END YEAR OF APPLE RUN
C     HHYR      -   END YEAR OF HOUSEHOLD HEADSHIP RATE
C                   CHANGE
C     CYCLE     -   LENGTH OF CYCLE (1 OR 5)
C     PROP      -   ARRAY 4 BY 10 CONTAINING PROPORTION
C                   CHANGE FROM BAYSE YEAR TO END OF FORECAST
C                   YEAR
C     HHINIT    -   BASE YEAR HOUSEHOLD HEADSHIP RATES
C     INTERP    -   NUMBER OF INTERPOLATIONS NECESSARY
C     HH9       -   ARRAY 4 BY 10 CONTAINING END YEAR OF
C                   RATE CHANGE HOUSEHOLD HEADSHIP RATES
C     RACES     -   NUMBER OF RACES
      DIMENSION  HHINIT(4,10),HH9(4,10),HH(4,10),PROP(4,10)
      INTEGER CYCLE,LENGTH,ENDYR,BASEYR,HHYR,INTERP,RACES,
     1IHH(4,10),IHHIT(4,10),COUNT,KK
C     READ PARAMETERS
      READ (5,200) BASEYR,ENDYR,HHYR,CYLE,RACES
200   FORMAT (3I4,2I2)
      COUNT = 0
      KRACES = RACES * 2
C     READ BASE YEAR HEADSHIP RATES
      DO 111 I = 1,4
      READ (5,205) (HHINIT(I,J),J=1,10)
      DO 111 J=1,10
111   IHHIT(I,J)=IFIX(1000*HHINIT(I,J))
C     READ PROPORTION CHANGE
      DO 666 I=1,4
666   READ (5,210) (PROP(I,J),J=1,10)
205   FORMAT (10F3.3)
210   FORMAT (10F4.3,32X)
      WRITE (6,215) BASEYR,ENDYR,HHYR,CYLE,RACES
215   FORMAT ('1 HOUSEHOLD HEADSHIP RATES',//,' BASE YEAR ',
     XI5,//,' END YEAR OF RUN ',I5,//,' END YEAR OF CHANGE ',
     X'FORECAST ',I5,///,' CYCLE LENGTH ',I2,//,' RACES ',I2)
C     COPY THE BASE YEAR RATES TO OUTPUT FILE
229   FORMAT (10I3)
220   FORMAT (10F3.3)
C     PRINT OUT INITIAL RATES
      DO 900 I = 1,4
      CALL TITLE (I,RACES)
      IF (I .GT. KRACES) GO TO 900
      WRITE (6,225) (HHINIT(I,J),J=1,10)
900   CONTINUE
C     COMPUTE END YEAR RATES
      DO 300 I=1,4
      DO 350 J=1,10
      HH9(I,J) = PROP(I,J) * HHINIT(I,J)
      IF (HH9(I,J) .GT. .999) HH9(I,J) = .999
```

```
 50   CONTINUE
C     PRINT THE END YEAR RESULTS FOR EACH VALUE OF I
C     FOR WHICH THERE IS MEANINGFUL DATA
      IF (I .GT. KRACES) GO TO 300
      IF (I .GT. 1) GO TO 222
      WRITE (6,221) ENDYR
221   FORMAT ('1  END YEAR HOUSEHOLD HEADSHIP RATES--YEAR = ',2X,I5)
227   CALL TITLE (I,RACES)
      WRITE (6,225) (HH9(I,J),J=1,10)
225   FORMAT (//,'  0-14',5X,F4.3,/,' 15-19',5X,F4.3,/,
     X          ' 20-24',5X,F4.3,/,' 25-29',5X,F4.3,/,
     X          ' 30-34',5X,F4.3,/,' 35-44',5X,F4.3,/,
     X          ' 45-54',5X,F4.3,/,' 55-64',5X,F4.3,/,
     X          ' 65-74',5X,F4.3,/,' 75+  ',5X,F4.3)
      GO TO 300
222   WRITE (6,228) ENDYR
228   FORMAT (//,'  END YEAR HOUSEHOLD HEADSHIP RATES--YEAR = ',2X,I5,/)
      GO TO 227
300   CONTINUE
C     THIS ROUTINE COMPUTES THE MIDYEAR RATES
      DO 540 KK=1975,2000,5
      IYEAR = KK
      IF (KK .GT. HHYR) IYEAR = HHYR
      DO 530 I=1,4
      DO 520 J=1,10
      IF (HHYR .GT. BASEYR) GO TO 555
      HH(I,J) = HHINIT(I,J)
      GO TO 444
555   HH(I,J) = ((IYEAR-BASEYR)/(HHYR-BASEYR)) *
     1(HH9(I,J)-HHINIT(I,J))  + HHINIT(I,J)
444   IHH(I,J) = IFIX(1000*HH(I,J))
520   CONTINUE
      WRITE (14,229) (IHH(I,J),J=1,10)
      IF (I .GT. KRACES) GO TO 530
      IF (I.GT. 1) GO TO 242
      WRITE (6,240) KK
240   FORMAT('1 ************ COMPUTED HOUSEHOLD HEADSHIP RATES FOR ',I5,
     X' **********',/)
247   CALL TITLE (I,RACES)
      WRITE (6,225) (HH(I,J),J=1,10)
      GO TO 530
242   WRITE (6,248) KK
248   FORMAT ('  ********** COMPUTED HOUSEHOLD HEADSHIP RATES FOR ',I5,
     X' **********',/)
      GO TO 247
530   CONTINUE
540   CONTINUE
      STOP
      END
      SUBROUTINE TITLE (I,IRACES)
C     PRINT SEX AND RACE ON REPORT
901   FORMAT(' WHITE MALES',//)
902   FORMAT(' WHITE FEMALES',//)
903   FORMAT(' NONWHITE MALES',//)
904   FORMAT(' NONWHITE FEMALES',//)
905   FORMAT(' MALES',//)
906   FORMAT(' FEMALES',//)
      IF(IRACES .EQ. 1) GO TO 100
      IF(I .EQ. 1) WRITE(6,901)
      IF(I .EQ. 2) WRITE(6,902)
```

```
      IF(I .EQ. 3) WRITE(6,903)
      IF(I .EQ. 4) WRITE(6,904)
      RETURN
100   CONTINUE
      IF(I .EQ. 1) WRITE(6,905)
      IF(I .EQ. 2) WRITE(6,906)
      RETURN
      END
```

LFPR: The Labor Force Participation Rate Generator

Purpose

Labor Force Participation Rate data generator interpolates from base year and end year figures, and allocates rates to the proper years. The generator receives as input: (1) projection timing parameters, (2) base year labor force participation rates, and (3) the expected percent change by the end of the projection.

Procedure

This generator produces labor force participation rates for each cycle of the projection period. In discussing this generator, distinction must be made between the labor force participation rate change projection period and the APPLE projection period. The rate change projection period is the length of time during which the rate change occurs, i.e. 1970 to 1990. After the rate of change end year, the labor force participation rates are assumed to remain constant at this end year level. The APPLE projection period, however, is the length of the total population projection model period, in this case, the year 2000.

Base year rates are read from the input file, stored, and copied to the print file. Rates applicable to the end of projection year are computed, using the percentage change figures as read from input and applying these to the base year labor force participation rates. These end of projection year rates are printed, along with the change factors.

To develop mid-forecast cycle rates, simple linear interpolation is used. Mid-cycle rates, and not end-cycle rates are of interest, and so instead of 1975, 1980, etc., interpolated values for labor rates are computed for mid 1972, mid 1977, etc. (Since the generator is adaptable to a one year cycle length, the calculations under such circumstances would produce labor force participation rates for mid 1970, mid 1971, etc.)

For those cycles falling outside of the projection period, the end of period rates apply.

Dictionary of Variables

BASE - Base year of APPLE run

END - End year of APPLE run

LFRC9 - End year of labor force rate change forecast

CYCLE - Output interval 1 or 5 years

BASELR - Labor force participation rates for the base year
an array of dimension (4,18)

PCHAN - Proportional change in the labor force participation rates to the end year period

an array of dimension (4,18)

LFPR9 - Labor force participation rates for the end year of rate change forecast

an array of dimension (4,18)

RACES - Number of races to be considered (1 or 2)
input to the TITLE Subroutine

LFPRC - Calculated values of labor force participation rates at the end of each cycle

an array of dimension (4,18)

KRACE - Dummy variables = RACES * SEXES(2) to control loops

FIGURE 7

LFPR PROGRAM FLOW

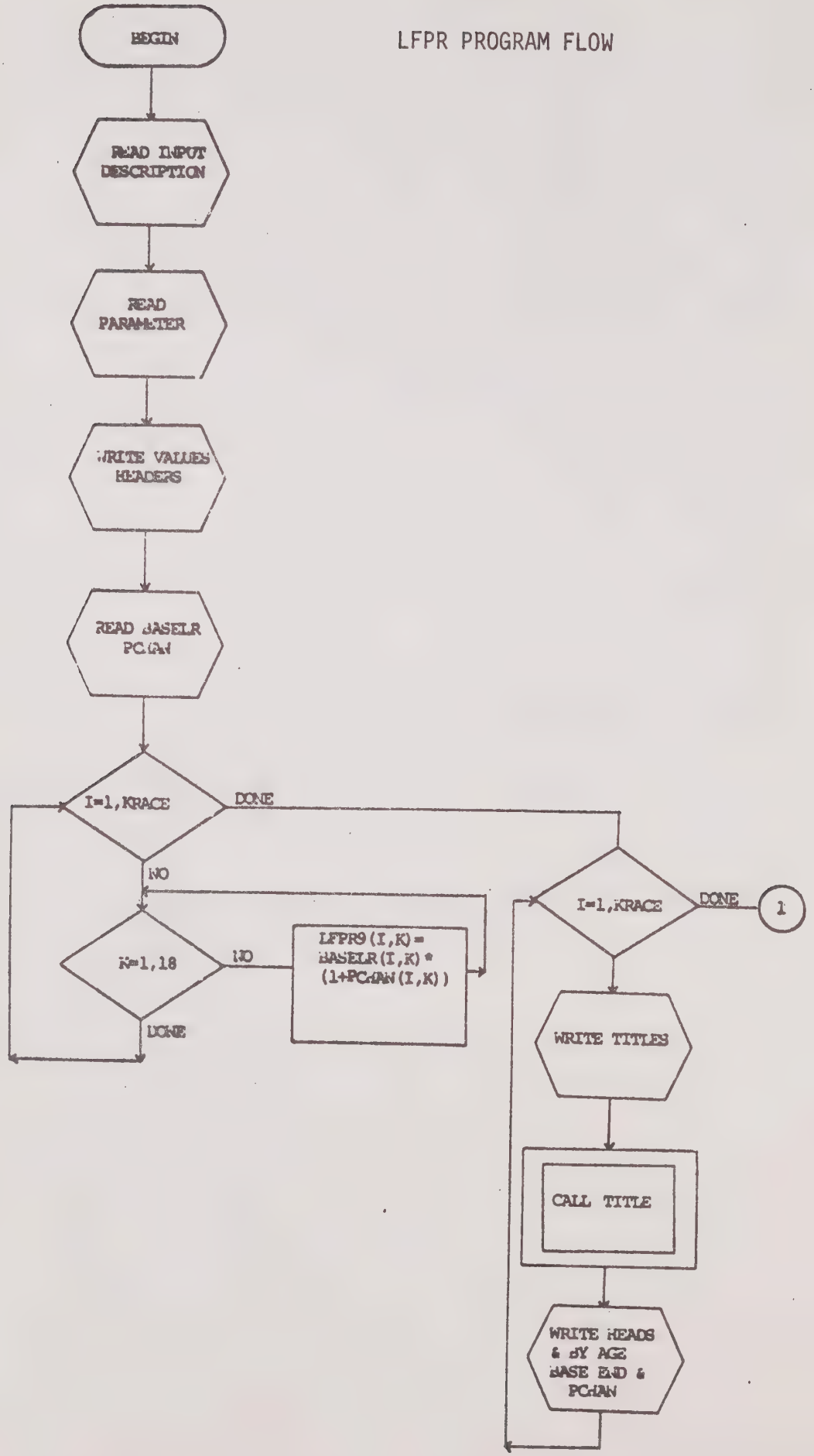
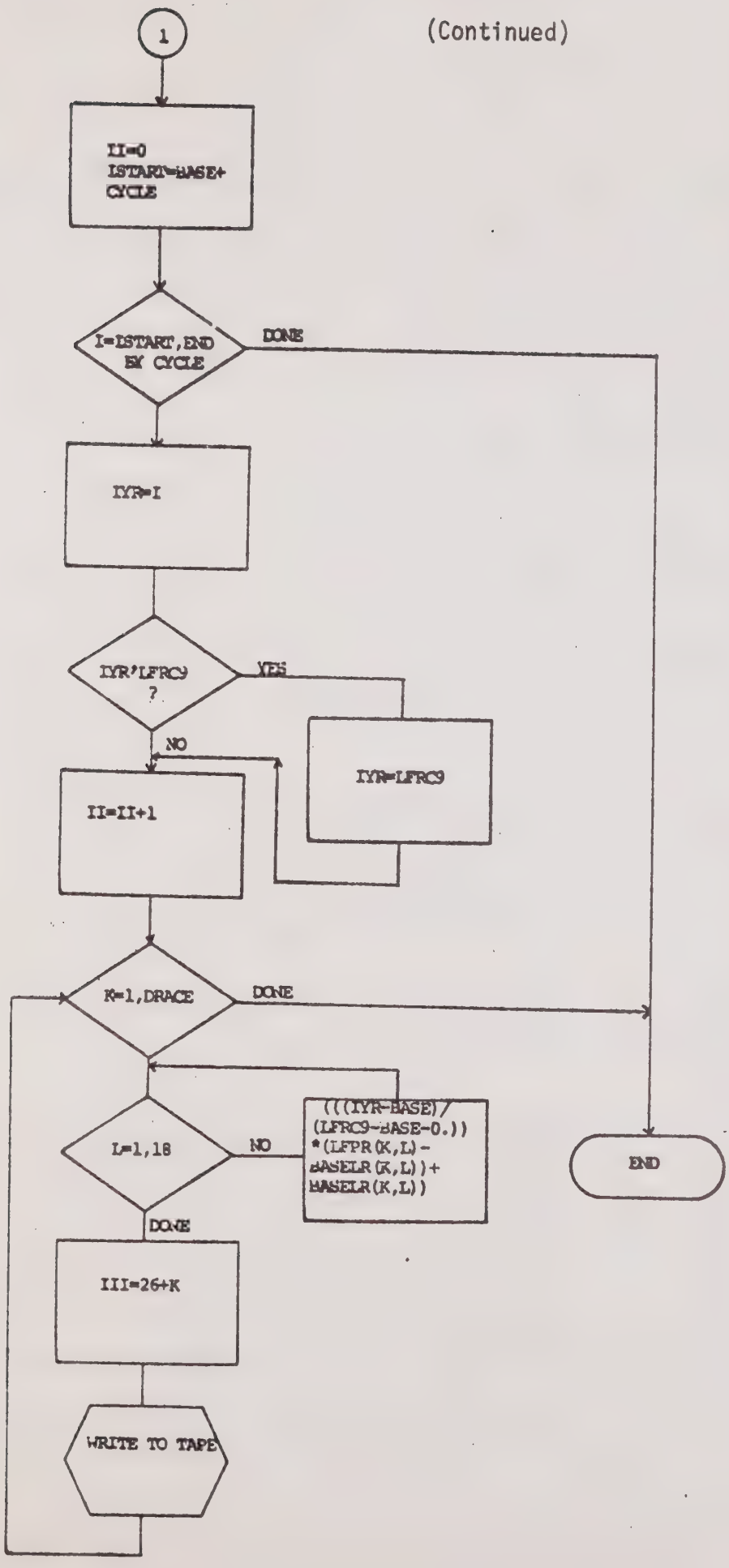


FIGURE 7

(Continued)



LFPR Program

Input Description

Input to the generator consists of a title card, a set of run parameters, base year labor force participation rates for white and nonwhite males and females, and percentage change figures for white and nonwhite males and females.

The format is as follows:

CARD	COLUMNS	DESCRIPTION
1		TITLE (20A4)
	1-80	Title
2		PARAMETER (3I4,2I2)
	1-4	Base year of APPLE run
	5-8	End year of APPLE run
	9-12	End year of rate change projection
	13-14	Cycle lenght
	15-16	Number of races, 1 or 2
3		BASE YEAR LABOR FORCE PARTICIPATION RATES (18I4)
	1-3	0-4 Labor force participation rate, white male
	4-6	5-9 Labor force participation rate, white male
	7-9	10-14 Labor force participation rate, white male
	10-12	15-19 Labor force participation rate, white male
	13-15	20-24 Labor force participation rate, white male

16-18	25-29 Labor force participation rate, white male
19-21	30-34 Labor force participation rate, white male
22-24	35-39 Labor force participation rate, white male
25-27	40-44 Labor force participation rate, white male
28-30	45-49 Labor force participation rate, white male
31-33	50-54 Labor force participation rate, white male
34-36	55-59 Labor force participation rate, white male
37-39	60-64 Labor force participation rate, white male
40-42	65-69 Labor force participation rate, white male
43-45	70-74 Labor force participation rate, white male
46-48	75-79 Labor force participation rate, white male
49-51	80-84 Labor force participation rate, white male
52-54	85+ Labor force participation rate, white male

4 BASE YEAR LABOR FORCE PARTICIPATION RATES, white females

Same as Card No. 3

5 BASE YEAR LABOR FORCE PARTICIPATION RATES, nonwhite males

Same as Card No. 3

6 BASE YEAR LABOR FORCE PARTICIPATION RATES, nonwhite females

Same as Card No. 3

CARD	COLUMNS	DESCRIPTION
7		FRACTION CHANGE IN LFPR, WHITE MALE
	1-4	0-4 Fraction change in LFPR, white male
	5-8	5-9 Fraction change in LFPR, white male
	9-12	10-14 Fraction change in LFPR, white male
	13-16	15-19 Fraction change in LFPR, white male
	17-20	20-24 Fraction change in LFPR, white male
	21-24	25-29 Fraction change in LFPR, white male
	25-28	30-34 Fraction change in LFPR, white male
	29-32	35-39 Fraction change in LFPR, white male
	33-36	40-44 Fraction change in LFPR, white male
	37-40	45-49 Fraction change in LFPR, white male
	41-44	50-54 Fraction change in LFPR, white male
	45-48	55-59 Fraction change in LFPR, white male
	49-52	60-64 Fraction change in LFPR, white male
	53-56	65-69 Fraction change in LFPR, white male
	57-60	70-74 Fraction change in LFPR, white male
	61-64	75-79 Fraction change in LFPR, white male
	65-68	80-84 Fraction change in LFPR, white male
	69-72	85+ Fraction change in LFPR, white male
8		FRACTION CHANGE IN LFPR, WHITE FEMALE
		Same as Card No. 7
9		FRACTION CHANGE IN LFPR, NONWHITE MALE
		Same as Card No. 7

10	FRACTION CHANGE IN LFPR, NONWHITE FEMALE
	Same as Card No. 7

Output Description

LPFR, generates file 12 which contains labor force participation rates for the number of cycles (indicated as an input parameter). The program also generates a printed report, indicating base year figures and percentage change in the rates, as well as the end year calculations.

The format for file 12 is as follows:

LABOR FORCE PARTICIPATION RATES

RECORD 1

CARD	COLUMN	DESCRIPTION
	1-3	0-4 Labor force participation rate, white male, cycle 1
	4-6	5-9 Labor force participation rate, white male, cycle 1
	7-9	10-14 Labor force participation rate, white male, cycle 1
	10-12	15-19 Labor force participation rate, white male, cycle 1
	13-15	20-24 Labor force participation rate, white male, cycle 1
	16-18	25-29 Labor force participation rate, white male, cycle 1
	19-21	30-34 Labor force participation rate, white male, cycle 1
	22-24	35-39 Labor force participation rate, white male, cycle 1

25-27	40-44 Labor force participation rates, white male, cycle 1
28-30	45-49 Labor force participation rates, white male, cycle 1
31-33	50-54 Labor force participation rates, white male, cycle 1
34-36	55-59 Labor force participation rates, white male, cycle 1
37-39	60-64 Labor force participation rates, white male, cycle 1
40-42	65-69 Labor force participation rates, white male, cycle 1
43-45	70-74 Labor force participation rates, white male, cycle 1
46-48	75-79 Labor force participation rates, white male, cycle 1
49-51	80-84 Labor force participation rates, white male, cycle 1
52-54	85+ Labor force participation rates, white male, cycle 1

RECORD 2 (SAME FORMAT AS ABOVE, WHITE FEMALES)

RECORD 3 (SAME FORMAT AS ABOVE, NONWHITE MALES)

RECORD 4 (SAME FORMAT AS ABOVE, NONWHITE FEMALES)

RECORD 5 THROUGH 24 HAVE THE SAME FORMAT AS 1 THROUGH 4

Technical Specifications

LFPR reads file 5, SYSIN, and writes the report to file 6, SYSOUT. The output data file is written to file 12. Below is the JCL (Job Contol Language) necessary to generate these files when executing LFPR:

```
//FT05F001   DD  DDNAME=SYSIN
//FT06F001   DD  SYSOUT=A
//FT12F001   DD  DSN=name,UNIT=unit type, SPACE=(TRK,(5,1),RLSE),
//           VOL=SER=Volname,DISP=(,CATLG),DCB=(LRECL=80,BLKSIZE=3520,
//           RECFM=FB)
//SYSIN      DD *
  data
/*
```

Timing and core requirements for execution of LFPR are:

```
CPU TIME     .021
REGION USED  38K
```

```
C     THIS IS LFPR.  IT GENERATES FILE 12 FOR APPLE.
C     PROGRAM TO GENERATE LABOR FORCE PARTICIPATION RATES
C     BASED ON INPUT OF BASE YEAR LFPR(S) AND INPUT OF FORECAST
C     PROPORTIONAL CHANGES IN THE RATES AT END YEAR OF RATE CHANGE
C     FORECAST METHOD USES SIMPLE LINEAR INTERPOLATION TO DERIVE RATES
C     BETWEEN BASE YEAR AND END YEAR OF RATE CHANGE FORECAST
C     RATES DO NOT CHANGE AFTER END YEAR OF RATE CHANGE FORECAST
      REAL REM(20),BASELR (4,18), PCHAN(4,18),LFPR9(4,18)
      INTEGER BASE,END,LFRC9,CYCLE,RACES,LFPRC(18)
C     BASE          -    BASE YEAR OF APPLE RUN
C     END           -    END YEAR OF APPLE RUN
C     LFRC9         -    END YEAR OF LABOR FOR RATE CHANGE FORECAST
C     CYCLE         -    '1' OR '5' YEAR CYCLES OF OUTPUT
C     BASELR        -    LABOR FORCE PARTICIPATION RATES (APPLE BASE YEAR)
C     PCHAN         -    PROPORTIONAL CHANGE IN LFPR(S)
C     LFPR9         -    LABOR FORCE PARTICIPATION RATES (END YEAR OF
C                        RATE CHANGE FORECAST)
C     RACES         -    NUMBER OF RACES (1 OR 2), CONTROLS PRINTING
C                        OF RACE 2
C     LFPRC         -    CALCULATED VALUES OF LABOR FORCE PARTICIPATION
C                        RATES AT END OF EACH CYCLE
C     READ INPUT DESCRIPTION
      READ(5,901) REM
901   FORMAT (20A4)
C     READ PARAMETER CARD
      READ(5,902) BASE,END,LFRC9,CYCLE,RACES
902   FORMAT (3I4,2I2)
      WRITE(6,905) REM,BASE,END,LFRC9,CYCLE,RACES
905   FORMAT ('1 LABOR FORCE PARTICIPATION RATES GENERATOR',49X,//
     X         ' INPUT-',2X,20A4,/,
     X         ' APPLE BASE YEAR -',I6,/,
     X         ' APPLE END YEAR -',I6,/,
     X         ' END YEAR OF RATE CHANGE FORECAST-',I6,/,
     X         ' YEARS PER CYCLE- ',I3,/,
     X         ' RACES- ',I3,//)
C     READ BASE YEAR LABOR FORCE PARTICIPATION RATES
      READ(5,903) ((BASELR(I,J),J=1,18),I=1,4)
903   FORMAT   (18F3.3)
C     READ PROPORTION CHANGE IN LABOR FORCE PARTICIPATION RATES
C     FROM APPLE BASE YEAR TO END YEAR OF RATE CHANGE FORECAST
      READ(5,904) ((PCHAN(I,J),J=1,18),I=1,4)
904   FORMAT   (18F4.3)
C     CALCULATE LABOR FORCE PARTICIPATION RATES AT END YEAR OF RATE
C     CHANGE FORECAST
      DO 100 I = 1,4
      DO 100 K = 1,18
      LFPR9(I,K) = BASELR(I,K) * PCHAN(I,K)
100   CONTINUE
C     PRINT OUT INPUT AND CALCULATED RATES
C
C
      DO 200 I=1,4
      IF (RACES .EQ. 1 .AND. I .GT.2) GO TO 200
      WRITE(6,910)
910   FORMAT ('1 LABOR FORCE PARTICIPATION RATES')
      CALL TITLE (I, RACES)
      WRITE(6,920) BASE, LFRC9
```

```
920   FORMAT ('  AGE ', 10X, I5, 10X,'PROPORTION',10X,I5,/,
     X        16X, 'LFPR', 13X, 'CHANGE', 12X, 'LFPR',//)
      DO 150 K= 1, 17
      I1=(5*K)-5
      I2 = I1 + 4
      WRITE(6,930) I1,I2, BASELR (I,K), PCHAN (I,K), LFPR9 (I,K)
930     FORMAT (1X,I2, '-', I2, 9X, F5.3, 13X, F5.3, 11X, F5.3)
150   CONTINUE
      WRITE(6,940) BASELR (I,18), PCHAN (I,18), LFPR9 (I,18)
940   FORMAT (' 85+', 11X, F5.3, 13X, F5.3, 11X, F5.3)
200   CONTINUE
      II = 0
C          CALCULATE AND WRITE OUT LABOR FORCE PARTICIPATION RATES AT
C     END OF EACH CYCLE
      ISTART = BASE + CYCLE
      DO 400 I = ISTART, END, CYCLE
      IYR = I
      IF (I YR .GT. LFRC9) IYR = LFRC9
      II = II+1
      DO 400 K= 1,4
      DO 350 L = 1,18
      LFPRC (L) = 1000 * (((IYR-BASE)/(LFRC9-BASE-0.)) * (LFPR9 (K,L)
     --BASELR(K,L)) + BASELR(K,L))
350   CONTINUE
      III = 26+K
      WRITE (12,960) LFPRC,II,III
960   FORMAT (18I3, 21X, I2, I2)
400   CONTINUE
      STOP
      END
      SUBROUTINE TITLE(I,IRACES)
C     PRINT SEX AND RACE ON REPORT
901   FORMAT(' WHITE MALES',//)
902   FORMAT(' WHITE FEMALES',//)
903   FORMAT(' NONWHITE MALES',//)
904   FORMAT(' NONWHITE FEMALES',//)
905   FORMAT(' MALES',//)
906   FORMAT(' FEMALES',//)
      IF(IRACES .EQ. 1) GO TO 100
      IF (I .EQ. 1)  WRITE(6,901)
      IF (I .EQ. 2) WRITE(6,902)
      IF (I .EQ. 3) WRITE(6,902)
      IF (I .EQ. 4) WRITE(6,904)
      RETURN
100   CONTINUE
      IF(I .EQ. 1) WRITE(6,905)
      IF(I .EQ. 2) WRITE(6,906)
      RETURN
      END
```

KGEN: The Constant Data Generator

Purpose

The KGEN, the Constant Data Generator, reproduces: (1) the base year data, and (2) constant assumption data, required for each cycle of APPLE. The program copies images from the input stream and prepares it for the read statements in the APPLE main program. Output from this program is meshed with output from the other generators FERT, SURV, LFPR, and HHEAD to form necessary input to each APPLE run.

Input Description

Base year data includes such information as: military dependent population, total labor force, all levels of school enrollments, total households, group quarters, distribution of births by sex, distribution of migrants by age and sex (and race), base year total population by age, race, sex, base year military population by age, race, sex, base year retirement migrants by age, race, sex, school participation rates by age, race, sex for all levels of education.

KGEN also requires one 72 character input record as a title input description to label the printed output.

Format for input:

CARD	COLUMNS	DESCRIPTION
1		TITLE
	1-72	Input description (18A4)
2		PARAMETERS (2I4, 2I2)
	1-4	Base year

CARD	COLUMNS	DESCRIPTION
2		PARAMETERS (continued)
	5-8	End year
	9-10	Cycle
	11-12	Cycles
3		MISCELLANEOUS BASE YEAR DATA (these data are used in printing the Summary Table only, and do not enter into any calculations)
	1-7	Total base year military dependent population
	8-14	Total base year civilian labor force
	15-21	Total base year school enrollment - Nursery
	22-28	Total base year school enrollment - Kindergarten
	29-35	Total base year school enrollment - Elementary
	36-42	Total base year school enrollment - High School
	43-49	Total base year school enrollment - College
	50-56	Total base year (occupied) households
	57-62	Total base year military group quarters population
	63-68	Total base year college dormitory group quarters population
	69-74	Total base year other group quarter population
4		FRACTION MALE BIRTHS (F3.3, 1X, F3.3)
	1-3	Fraction of total births which are male-white
	5-7	Fraction of total births which are male-nonwhite
5		DISTRIBUTION OF EMPLOYMENT RELATED MIGRANTS, White males (ratio of migrants in each age group to total migrants) (13F 6.4)
	1-6	Fraction of migrants 0-4 years
	7-12	Fraction of migrants 5-9 years

CARD	COLUMNS	DESCRIPTION
5		DISTRIBUTION OF EMPLOYMENT RELATED MIGRANTS (continued)
	13-18	Fraction of migrants 10-14 years
	19-24	Fraction of migrants 15-19 years
	25-30	Fraction of migrants 20-24 years
	31-36	Fraction of migrants 25-29 years
	37-42	Fraction of migrants 30-34 years
	43-48	Fraction of migrants 35-39 years
	49-54	Fraction of migrants 40-44 years
	55-60	Fraction of migrants 45-49 years
	61-66	Fraction of migrants 50-54 years
	67-72	Fraction of migrants 55-59 years
	73-78	Fraction of migrants 60-64 years
6		DISTRIBUTION OF EMPLOYMENT RELATED MIGRANTS, white males (5F 6.4)
	1-6	Fraction of migrants 65-69 years
	7-12	Fraction of migrants 70-74 years
	13-18	Fraction of migrants 75-79 years
	19-24	Fraction of migrants 80-84 years
	25-30	Fraction of migrants 85 and over years
7		DISTRIBUTION OF EMPLOYMENT RELATED MIGRANTS, white females (Ratio of migrants in age group to total migrants)
		Same as Card No. 5
8		DISTRIBUTION OF EMPLOYMENT RELATED MIGRANTS, white females
		Same as Card No. 6

CARD	COLUMNS	DESCRIPTION
9		DISTRIBUTION OF EMPLOYMENT RELATED MIGRANTS, nonwhite males (Ratio of migrants in age group to total migrants)
		Same as Card No. 5
10		DISTRIBUTION OF EMPLOYMENT RELATED MIGRANTS, nonwhite males
		Same as Card No. 6
11		DISTRIBUTION OF EMPLOYMENT RELATED MIGRANTS, nonwhite females (Ratio of migrants in age group to total migrants)
		Same as Card No. 5
12		DISTRIBUTION OF EMPLOYMENT RELATED MIGRANTS, nonwhite females
		Same as Card No. 6
13		BASE YEAR POPULATION, white males, ages 0-44 (total base year population, including military in-service and dependents) (9I7)
	1-7	0-4 base year population, white males
	8-14	5-9 base year population, white males
	15-21	10-14 base year population, white males
	22-28	15-19 base year population, white males
	29-35	20-24 base year population, white males
	36-42	25-29 base year population, white males
	43-49	30-34 base year population, white males
	50-56	35-39 base year population, white males
	57-63	40-44 base year population, white males

CARD	COLUMNS	DESCRIPTION
14		BASE YEAR POPULATION (continued)
	1-7	45-49 base year population, white males
	8-14	50-54 base year population, white males
	15-21	55-59 base year population, white males
	22-28	60-64 base year population, white males
	29-35	65-69 base year population, white males
	36-42	70-74 base year population, white males
	43-49	75-79 base year population, white males
	50-56	80-84 base year population, white males
	57-63	85 and over base year population, white males
15		BASE YEAR POPULATION, white females, ages 0-44
		Same as Card No. 13
16		BASE YEAR POPULATION, white females, ages 45-85 and over
		Same as Card No. 14
17		BASE YEAR POPULATION, nonwhite males, ages 0-44
		Same as Card No. 13
18		BASE YEAR POPULATION, nonwhite males, ages 45-85 and over
		Same as Card No. 14
19		BASE YEAR POPULATION, nonwhite females, ages 0-44
		Same as Card No. 13
20		BASE YEAR POPULATION, nonwhite females, ages 45-85 and over
		Same as Card No. 14

CARD	COLUMNS	DESCRIPTION
21		BASE YEAR MILITARY POPULATION (includes both in-service and dependent) white males, ages 0-44
	1-7	0-4 base year special population, white males
	8-14	5-9 base year special population, white males
	15-21	10-14 base year special population, white males
	22-28	15-19 base year special population, white males
	29-35	20-24 base year special population, white males
	36-42	25-29 base year special population, white males
	43-49	30-34 base year special population, white males
	50-56	35-39 base year special population, white males
	57-63	40-44 base year special population, white males
22		BASE YEAR MILITARY POPULATION (includes both in-service and dependent), white males, ages 45-85 and over (9I7)
	1-7	45-49 base year special population, white males
	8-14	50-54 base year special population, white males
	15-21	55-59 base year special population, white males
	22-28	60-64 base year special population, white males
	29-35	65-69 base year special population, white males
	36-42	70-74 base year special population, white males
	43-49	75-79 base year special population, white males
	50-56	80-84 base year special population, white males
	57-63	85 and over base year special population, white males
23		BASE YEAR MILITARY POPULATION (includes both in-service and dependent), white females, ages 0-44
		Same as Card No. 21

CARD	COLUMNS	DESCRIPTION
24		BASE YEAR MILITARY POPULATION (includes both in-service and dependent), white females, ages 45-85 and over
		Same as Card No. 22
25		BASE YEAR MILITARY POPULATION (includes both in-service and dependent), nonwhite males, ages 0-44
		Same as Card No. 21
26		BASE YEAR MILITARY POPULATION (includes both in-service and dependent), nonwhite males, ages 45-85 and over
		Same as Card No. 22
27		BASE YEAR MILITARY POPULATION (includes both in-service and dependent), nonwhite females ages 0-44
		Same as Card No. 21
28		BASE YEAR MILITARY POPULATION (includes both in-service and dependent), nonwhite females, ages 45-85.
		Same as Card No. 22
29		MISCELLANEOUS BASE YEAR DATA (5X, 5I10)
	6-15	Base year basic employment
	16-25	Base year (occupied) households
	26-35	Base year school enrollment
	36-45	Base year state college enrollment
	46-55	Base year population
30		MILITARY DEPENDENT POPULATION, white males, ages 0-44 (9I7)
	1-7	0-4 special population included in labor force, white males
	8-14	5-9 special population included in labor force, white males

CARD	COLUMNS	DESCRIPTION
30		MILITARY DEPENDENT POPULATION (continued)
	15-21	10-14 special population included in labor force, white males
	22-28	15-19 special population included in labor force, white males
	29-35	20-24 special population included in labor force, white males
	36-42	25-29 special population included in labor force, white males
	43-49	30-34 special population included in labor force, white males
	50-56	35-39 special population included in labor force, white males
	57-63	40-44 special population included in labor force, white males
31		MILITARY DEPENDENT POPULATION, white males, ages 45-85 and over (9I7)
	1-7	45-49 special population included in labor force, white males
	8-14	50-54 special population included in labor force, white males
	15-21	55-59 special population included in labor force, white males
	22-28	60-64 special population included in labor force, white males
	29-35	65-69 special population included in labor force, white males
	36-42	70-74 special population included in labor force, white males
	43-49	75-79 special population included in labor force, white males
	50-56	80-84 special population included in labor force, white males
	57-63	85 and over special population included in labor force, white males

CARD	COLUMNS	DESCRIPTION
32		MILITARY DEPENDENT POPULATION, white females, ages 0-44
		Same as Card No. 30
33		MILITARY DEPENDENT POPULATION, white females, ages 45-85 and over
		Same as Card No. 31
34		MILITARY DEPENDENT POPULATION, nonwhite males, ages 0-44
		Same as Card No. 30
35		MILITARY DEPENDENT POPULATION, nonwhite males, ages 45-85 and over
		Same as Card No. 31
36		MILITARY DEPENDENT POPULATION FORCE, nonwhite females, ages 0-44
		Same as Card No. 30
37		MILITARY DEPENDENT POPULATION FORCE, nonwhite females, ages 45-85 and over
		Same as Card No. 31
38		SCHOOL PARTICIPATION RATES NURSERY, white males and females, ages 0-34, at end of interval (fraction of civilian population in age-race-sex group attending nursery school) (7I3, 2X, 7I3)
	1-3	0-4 school participation rate - nursery - white males
	4-6	5-9 school participation rate - nursery - white males
	7-9	10-14 school participation rate - nursery - white males
	10-12	15-19 school participation rate - nursery - white males
	13-15	20-24 school participation rate - nursery - white males
	16-18	25-29 school participation rate - nursery - white males

CARD	COLUMNS	DESCRIPTION
38		SCHOOL PARTICIPATION RATES NURSERY (continued)
	19-21	30-34 school participation rate - nursery - white males
	22-23	Blank
	24-26	0-4 school participation rate - nursery - white females
	27-29	5-9 school participation rate - nursery - white females
	30-32	10-14 school participation rate - nursery - white females
	33-35	15-19 school participation rate - nursery - white females
	36-38	20-24 school participation rate - nursery - white females
	39-41	25-29 school participation rate - nursery - white females
	42-44	30-34 school participation rate - nursery - white females
39		SCHOOL PARTICIPATION RATES - NURSERY, nonwhite males and females, ages 0-34
		Same as Card No. 38
40		SCHOOL PARTICIPATION RATES - KINDERGARTEN, white males and females, ages 0-34
		Same as Card No. 38
41		SCHOOL PARTICIPATION RATES - KINDERGARTEN, white males and females, ages 0-34
		Same as Card No. 38
42		SCHOOL PARTICIPATION RATES - ELEMENTARY, white males and females, ages 0-34
		Same as Card No. 38
43		SCHOOL PARTICIPATION RATES - ELEMENTARY, nonwhite males and females, ages 0-34
		Same as Card No. 38

CARD	COLUMNS	DESCRIPTION
44		SCHOOL PARTICIPATION RATES - HIGH SCHOOL, white males and females, ages 0-34
		Same as Card No. 38
45		SCHOOL PARTICIPATION RATES - HIGH SCHOOL, nonwhite males and females, ages 0-34
		Same as Card No. 38
46		SCHOOL PARTICIPATION RATES - COLLEGE, white males and females, ages 0-34
		Same as Card No. 38
47		SCHOOL PARTICIPATION RATES - COLLEGE, nonwhite males and females, ages 0-34
		Same as Card No. 38
48		FRACTION IN GROUP QUARTERS - MILITARY BARRACKS, white, nonwhite males, ages 0-75 and over, (fraction of military in-service population in age group living in military barracks) (1013, 2X, 1013)
	1-3	0-14 fraction of military population in group quarters, white males
	4-6	15-19 fraction of military population in group quarters, white males
	7-9	20-24 fraction of military population in group quarters, white males
	10-12	25-29 fraction of military population in group quarters, white males
	13-15	30-34 fraction of military population in group quarters, white males
	16-18	35-44 fraction of military population in group quarters, white males
	19-21	45-54 fraction of military population in group quarters, white males

CARD	COLUMNS	DESCRIPTION
48		FRACTION IN GROUP QUARTERS - MILITARY BARRACKS (continued)
	22-24	55-64 fraction of military population in group quarters, white males
	25-27	65-74 fraction of military population in group quarters, white males
	28-30	75 and over fraction of military population in group quarters, white males
	31-32	Blank
	33-35	0-14 fraction of military population in group quarters, nonwhite males
	36-38	15-19 fraction of military population in group quarters, nonwhite males
	39-41	20-24 fraction of military population in group quarters, nonwhite males
	42-44	25-29 fraction of military population in group quarters, nonwhite males
	45-47	30-34 fraction of military population in group quarters, nonwhite males
	48-50	35-44 fraction of military population in group quarters, nonwhite males
	51-53	45-54 fraction of military population in group quarters, nonwhite males
	54-56	55-64 fraction of military population in group quarters, nonwhite males
	57-59	65-74 fraction of military population in group quarters, nonwhite males
	60-62	75 and over fraction of military population in group quarters, nonwhite males
49		FRACTION GROUP QUARTERS - COLLEGE DORMITORIES, white males and females (fraction of college enrollment in each age-race-sex group living in dormitories) (10I3, 2X, 10I3)
	1-3	0-14 fraction of college enrollment in dormitories, white males

CARD	COLUMNS	DESCRIPTION
49		FRACTION GROUP QUARTERS - COLLEGE DORMITORIES (continued)
	4-6	15-19 fraction of college enrollment in dormitories, white males
	7-9	20-24 fraction of college enrollment in dormitories, white males
	10-12	25-29 fraction of college enrollment in dormitories, white males
	13-15	30-34 fraction of college enrollment in dormitories, white males
	16-18	35-44 fraction of college enrollment in dormitories, white males
	19-21	45-54 fraction of college enrollment in dormitories, white males
	22-24	55-64 fraction of college enrollment in dormitories, white males
	25-27	65-74 fraction of college enrollment in dormitories, white males
	28-30	75 and over fraction of college enrollment in dormitories, white males
	31-32	Blank
	33-35	0-14 fraction of college enrollment in dormitories, white females
	36-38	15-19 fraction of college enrollment in dormitories, white females
	39-41	20-24 fraction of college enrollment in dormitories, white females
	42-44	25-29 fraction of college enrollment in dormitories, white females
	45-47	30-34 fraction of college enrollment in dormitories, white females
	48-50	35-44 fraction of college enrollment in dormitories, white females
	51-53	45-54 fraction of college enrollment in dormitories, white females

CARD	COLUMNS	DESCRIPTION
49		FRACTION GROUP QUARTERS - COLLEGE DORMITORIES (continued)
	54-56	55-64 fraction of college enrollment in dormitories, white females
	57-59	65-74 fraction of college enrollment in dormitories, white females
	60-62	75 and over fraction of college enrollment in dormitories, white females
50		FRACTION GROUP QUARTERS - COLLEGE DORMITORIES, nonwhite males and females
		Same as Card No. 49
51		FRACTION GROUP QUARTERS - OTHER, white males and females (fraction of civilian population living in other group quarters)
		Same as Card No. 49
52		FRACTION GROUP QUARTERS - OTHER, nonwhite males and females
		Same as Card No. 49
53		MILITARY IN-SERVICE POPULATION, white males, ages 0-44 and over (Armed Forces) (9I7)
	1-7	0-4 special population not included in labor force, white males
	8-14	5-9 special population not included in labor force, white males
	15-21	10-14 special population not included in labor force, white males
	22-28	15-19 special population not included in labor force, white males
	29-35	20-24 special population not included in labor force, white males
	36-42	25-29 special population not included in labor force, white males

CARD	COLUMNS	DESCRIPTION
53		MILITARY IN-SERVICE POPULATION (continued)
	43-49	30-34 special population not included in labor force, white males
	50-56	35-39 special population not included in labor force, white males
	57-63	40-44 special population not included in labor force, white males
54		MILITARY IN-SERVICE POPULATION, white males, ages 45-85 and over (Armed Forces) (9I7)
	1-7	45-49 special population not included in labor force, white males
	8-14	50-54 special population not included in labor force, white males
	15-21	55-59 special population not included in labor force, white males
	22-28	60-64 special population not included in labor force, white males
	29-35	65-69 special population not included in labor force, white males
	36-42	70-74 special population not included in labor force, white males
	43-49	75-79 special population not included in labor force, white males
	50-56	80-84 special population not included in labor force, white males
	57-63	85 and over special population not included in labor force, white males
55		MILITARY IN-SERVICE POPULATION, white females, ages 0-44
		Same as Card No. 53
56		MILITARY IN-SERVICE POPULATION, white females, ages 45-85 and over
		Same as Card No. 54

CARD	COLUMNS	DESCRIPTION
57		MILITARY IN-SERVICE POPULATION, nonwhite males, ages 0-44
		Same as Card No. 53
58		MILITARY IN-SERVICE POPULATION, nonwhite males, ages 45-85 and over
		Same as Card No. 54
59		MILITARY IN-SERVICE POPULATION, nonwhite females, ages 0-44
		Same as Card No. 53
60		MILITARY IN-SERVICE POPULATION, nonwhite females, ages 45-85 and over
		Same as Card No. 54

Output Description

KGEN produces three output files: (1) a printed report that lists the APPLE run parameters, (2) a file containing one copy of the base year data for APPLE, (3) a file of constant assumption data for APPLE.

The report lists the paramters, base year, end year, and cycle length with appropriate labels.

Eighty character fixed block records are produced from the input data. The base year file is formatted exactly as the input data. The constant data is copied once to another file to be recycled within the APPLE program as required.

Dictionary of Variables

Array A -set of base year data

Array B -set of constant data

Baseyr	-base year of APPLE run
Endyr	-end year of APPLE run
CYCLE	-length of cycle, 1 or 5 years

Technical Specifications

KGEN reads file 5, SYSIN, and writes the report to file 6, SYSOUT. The two output data fies are written to files 11 and 15. Below is the format of the JCL (Job Control Language) necessary to generate these files when executing KGEN:

```
//FT05F001  DD DDNAME=SYSIN
//FT06F001  DD SYSOUT=A
//FT11F001  DD DSN=name,UNIT=unit type,SPACE=(TRK,(3,1)RLSE),
//          VOL=SER=volname,DISP=(,CATLG),DCB=(LRECL=80,BLKSIZE=3520,
//          RECFM=FB)
//FT15F001  DD DSN=name,UNIT=unit type,SPACE=(TRK,(3,1),RLSE),
//          VOL=SER=volname,DISP=(,CATLG),DCB=(LRECL=80,BLKSIZE=3520
//          RECFM=FB)
//SYSIN     DD *
  data
/*
```

Timing and core requirements for execution of KGEN are

CPU TIME	.011
REGION USED	48K

```
C
C       KGEN IS A VERY SIMPLE PROGRAM THAT READS THE INPUT STREAM
C       AND COPIES IT TO ONE OF TWO OUTPUT FILES
C       BASE YEAR DATA IS A SERIES OF STATISTICS REPRESENTING BASE YEAR
C       SITUATION
C       CONSTANT DATA DOCUMENTS DISTRUBUTION OF MILITARY  AND
C       DEPENDENTS IN THE POPULATION, SCHOOL PARTICIPATION RATES, ETC.
C       THESE RELATIONSHIPS ARE ASSUMED CONSTANT FOR THE PROJECTION PERIOD
C       OUTPUT FILES ARE 11(BASE YEAR) AND 15(CONSTANT DATA)
        REAL A(27,20),B(31,18)
        INTEGER BASEYR,ENDYR,CYCLE,CYCLES,TIMES
C       CYCLE      -    CYCLE LENGTH (EITHER 1 OR 5)
C       CYCLES     -    NUMBER OF CYCLES IN PROJECTION PERIOD
C                       = (ENDYR-BASEYR)/CYCLE
C       BASEYR     -    BASE YEAR OF APPLE RUN
C       ENDYR      -    END YEAR OF APPLE RUN
C       ARRAY A    -    CONTAINS BASE YEAR DATA
C                  -    THIS ARRAY IS READ AND WRITTEN ONCE
C       ARRAY B    -    CONTAINS CONSTANT DATA
C                  -    THIS ARRAY IS READ AND WRITTEN ONCE
        READ (5,900) BASEYR,ENDYR,CYCLE,CYCLES
900     FORMAT (2I4,2I2)
        WRITE (6,899) BASEYR,ENDYR,CYCLE,CYCLES
899     FORMAT ('1  CONSTANT GENERATOR  ',//,' BASE YEAR = ',
       XI4,/,' END YEAR = ',I4,/' CYCLE LENGTH = ',I2,
       X/,' TOTAL NUMBER OF CYCLES = ',I2)
C       LOOP 1
C       THIS LOOP READS THE BASE YEAR DATA FOUND IN THE 27 ROWS OF ARRAY A
        DO 100 I = 1,27
        READ (5,901) (A(I,J),J=1,20)
901     FORMAT (20A4)
        WRITE (11,902) (A(I,J),J=1,20)
902     FORMAT (20A4)
100     CONTINUE
C
C       LOOP 2
C       THIS LOOP READS THE CONSTANT DATA B(I,J) NEEDED FOR EACH CYCLE
        DO 600 I = 1,31
        READ (5,904,ERR=200) (B(I,J),J=1,18)
904     FORMAT (18A4)
        GO TO 500
  200   WRITE(6,401) I
  401   FORMAT('0READ ERROR OCCURRED - NUMBER OF B CARDS =',I4)
        STOP
500     WRITE (15,905) (B(I,J),J=1,18),I
905     FORMAT (18A4,2X,I2,2X)
600     CONTINUE
        ENDFILE 11
        ENDFILE 15
        STOP
        END
```

```
C
C     PROGRAM TO GENERATE SURVIVAL RATES BASED
C     ON INPUT OF BASE YEAR DEATH RATES AND
C     FORECAST OF PROPORTION OF DEATH RATE
C     REMAINING AT A FUTURE DATE
C
      REAL REM(20),BASEDR(4,19),CHANDR(4,19),BASDR(19),DR9T(19)
      REAL DR9(4,19),AGE(38),SRATE0(4,19), CAPL(4,19)
      REAL SRATE9(4,19),SRAT0(19),SRAT9(19),CAPLT(19)
      INTEGER BASE, END, DRC0, DRC9, CYCLE, RACES, YEAR
      INTEGER SRATEC (4,19), SRATES(87)
      DATA SRATE0/76*0.0/
      DATA SRATE9/76*0.0/
C     READ IN AGE CATEGORY LABELS
      DATA AGE/3H  0,3H-1 ,3H  1,3H-4 ,3H  5,3H-9 ,3H 10,3H-14,
     1         3H 15,3H-19,3H 20,3H-24,3H 25,3H-29,3H 30,3H-34,
     2         3H 35,3H-39,3H 40,3H-44,3H 45,3H-49,3H 50,3H-54,
     3         3H 55,3H-59,3H 60,3H-64,3H 65,3H-69,3H 70,3H-74,
     4         3H 75,3H-79,3H 80,3H-84,3H 85,3H-+  /
C     BASE          -    BASE YEAR OF APPLE RUN
C     END           -    ENDYEAR OF APPLE RUN
C     DRC0          -    BASE YEAR OF DEATH RATE CHANGE FORECAST
C     DRC9          -    ENDYEAR OF DEATH RATE CHANGE FORECAST
C     CYCLE         -    "1" OR "5" YEAR CYCLES OF OUTPUT
C     RACES         -    "1" OR "2" RACES CALCULATED
C     BASEDR        -    APPLE BASE YEAR DEATH RATES
C     DRC           -    ABBREVIATION FOR DEATH RATE CHANGE
C     CHANDR        -    PROPORTION CHANGE IN DEATH RATE
C     DR9           -    DRC ENDYEAR DEATH RATES
C     AGE           -    AGE INTERVALS
C     SRATE0(I)     -    SURVIVAL RATE FROM AGE I TO AGE I PLUS N(BASE
C                        YEAR
C     CAPL(I)       -    CAPITOL L, STATIONATY POPULATION AT AGE I
C     SRATE9(I)     -    SURVIVAL RATES AT ENDOF DRC FORECAST
C     SRATEC        -    SURVIVAL RATES FOR 5 YEAR CYCLE
C     YEAR          -    ENDYEAR OF CYCLE
C     SRATES        -    SURVIVAL RATES FOR SINGLE YEARS
C     READ RUN DESCRIPTION
      READ (5,900) REM
      WRITE (6,901) REM
900   FORMAT (20A4)
901   FORMAT ('1','SURVIVAL RATE GENERATOR FOR APPLE',//,1X,
     * 20A4,/)
C     READ DESCRIPTION OF BASE YEAR DEATH RATES
      READ (5,900) REM
      WRITE (6,902) REM
902   FORMAT (' ',20A4,/)
C     READ DESCRIPTION OF CHANGE IN DEATH RATES
      READ (5,900) REM
      WRITE (6,902) REM
C     READ PARAMETER CARD
      READ (5,903) BASE, END, DRC9, CYCLE,  RACES, DRC0
903   FORMAT (3I4,2I2,I4)
      WRITE (6,904) BASE, END, DRC0, DRC9, CYCLE, RACES
904   FORMAT (' RUN PARAMETERS',/,' APPLE BASE',
     X' YEAR-',I6,/,' APPLE ENDYEAR-', I6,/,
     X' DRC BASE YEAR-', I6,/,' DRC ENDYEAR',I6,/,
```

```
     X' YEARS PER CYCLE-',I3,/,' RACES-',I3)
C      READ APPLE BASE YEAR DEATH RATES
       READ (5,905)  ((BASEDR(I,J),J=1,19),I=1,4)
905    FORMAT (12F6.6,/,7F6.6)
C      READ PROPORTION OF CHANGE INDEATH RATES
C      BETWEEN BASE YEAR OF DRC FORECAST AND END
C      YEAR OF DRC FORECAST
       READ (5,908) ((CHANDR(I,J),J=1,19),rol,4)
908    FORMAT (19F3.3)
C      APPLY PROPORTIONAL CHANGE WHICH OCCURS DURING FORECAST TO APPLE
C      BASE YEAR DEATH RATES TO DERIVE DRC ENDYEAR DEATH RATES.
C      CALCULATE DEATH RATES AT ENDOF DRC FORECAST YEAR
       DO 100  I=1,4
       IF (RACES .EQ. 1 .AND. I .GT. 2) GO TO 100
       WRITE(6,909)
909    FORMAT(' INPUT AND CALCULATED VALUES',//)
       WRITE(6,910) BASE,DRC0,DRC9,BASE,DRC9,DRC9
910    FORMAT('  AGE   ',10X,I5,10X,I5,'-',I4,10X,I5,'-',I4,10X,I5,/,
      1 15X,'DEATH',10X,'PROPORTION',10X,'PROPORTION',10X,'DEATH',/,
      2 15X,'RATES',10X,'  CHANGE  ',10X,'  CHANGE  ',10X,'RATES',//)
       CALL TITLE (I,RACES)
       DO 90  K=1,19
       IF (DRC9-END) 701,701,700
700    A1 = ((DRC9-END)/(DRC9-DRC0)) * CHANDR(I,K)
       DR9(I,K) = BASEDR(I,K) * A1
REPLACE
       WRITE (6,915) AGE(2*K-1),AGE(2*K),BASEDR(I,K),CHANDR(I,K),A1,
      1 DR9(I,K)
       GO TO 705
701    DR9(I,K) = BASEDR(I,K)  * CHANDR(I,K)
C      PRINT OUT INPUT AND CALCULATED VALUES
       IF (RACES .EQ. 1 .AND. I .GT. 2) GO TO 90
       WRITE (6,915) AGE(2*K-1),AGE(2*K),BASEDR(I,K),CHANDR(I,K),
      1 CHANDR(I,K), DR9(I,K)
705    CONTINUE
915    FORMAT(1X,2A3,7X,4(F10.5,10X))
90     CONTINUE
100    CONTINUE
       DO 200 I=1,4
       IF(RACES .EQ. 1 .AND. I .GT. 2) GO TO 200
       WRITE (6,920) BASE
920    FORMAT ('1LIFE TABLE FOR BASE YEAR',I5,/)
       CALL TITLE (I,RACES)
C      CALCULATE LIFE TABLE VALUES FOR APPLE BASE YEAR
C      THIS NEXT STEP STRIPS ONE ROW AT A TIME
       DO 222 J = 1,19
       BASDR(J)=BASEDR(I,J)
222    CONTINUE
       CALL LIFE(BASDR,SRAT0,CAPLT)
       DO 444 J=1,19
       SRATE0(I,J) = SRAT0(J)
       CAPL(I,J) = CAPLT(J)
444    CONTINUE
       IF (CYCLE .EQ. 1) SRATE0(I,1) = CAPL(I,1) / 100000.
C      ELIMINATE SRATE(2),   MAKE SRATE(18) AND SRATE(19) THE SAME
       DO 150 K= 2,18
       SRATE0(I,K) = SRATE0(I,K+1)
150    CONTINUE
200    CONTINUE
C      CALCULATE LIFE TABLE VALUES FOR DRC ENDYEAR OF FORECAST
```

```
      DO 300 I=1,4
      IF(RACES .EQ. 1 .AND. I .GT. 2) GO TO 300
      WRITE (6,930) DRC9
930   FORMAT('1LIFE TABLE FOR END OF DEATH RATE CHANGE FORECAST YEAR',
     X I5,/)
      CALL TITLE (I,RACES)
C     THIS NEXT STEP STRIPS ONE ROW AT A TIME
      DO 333 J=1,19
      DR9T(J) = DR9(I,J)
333   CONTINUE
      CALL LIFE(DR9T,SRAT9,CAPLT)
      DO 555 J = 1,19
      SRATE9(I,J) = SRAT9(J)
      CAPL(I,J) = CAPLT(J)
555   CONTINUE
      IF (CYCLE .EQ. 1) SRATE9(I,1) = CAPL(I,1) / 100000.
C     ELIMINATE SRATE(2),   MAKE SRATE(18) AND SRATE(19) THE SAME
      DO 250 K=2,18
      SRATE9(I,K) = SRATE9(I,K+1)
250   CONTINUE
300   CONTINUE
C     PRODUCE MID INTERVAL SURVIVAL RATES FOR EACH 5 YEAR CYCLE
C     USES LINEAR INTERPOLATION
      II = 0
      I5 = BASE + 5
      DO 400 YEAR = I5,DRC9,5
      II = 22 + 1
      YEARC = YEAR
      IF(YEAR .GT. DRC9) YEARC = DRC9 + 2.5
      DO 390 I=1,4
      DO 340 K=1,19
      SRATEC(I,K) = 100000.*(SRATE0(I,K)+((YEARC-2.5-BASE)/
     1 (DRC9-BASE))*(SRATE9(I,K) - SRATE0(I,K)))  + .5
340   CONTINUE
      IF (CYCLE .EQ. 5) GO TO 385
C     PRODUCE SINGLE YEAR RATES FROM MID INTERVAL RATES
      SRATES(1) = ((SRATEC(I,1)/100000.)**.2) *100000.  +.5
      DO 350 K2 = 2,19
      K3 = K2 * 5 -8
      SRATES(K3) = ((SRATEC(I,K2) / 100000.)**.2) * 100000.  +.5
350   CONTINUE
      DO 360 K2 = 2,18
      DO 360 K3 = 1,4
      K4 = K2*5 -8
      SRATES(K4 + K3) = SRATES(K4)
360   CONTINUE
      DO 370 K2 = 1,5
      IF(K2 .EQ. 1) GO TO 366
      DO 365 K3 = 2,86
      SRATES(88-K3) = SRATES(87-K3)
365   CONTINUE
      SRATES(1) = ((SRATEC(I,1)/100000.)**(1./(6.-K2)))*100000. +.5
366   CONTINUE
      I1 = 20 * I - 10
      WRITE (13,980) (SRATES(L),L= 1,14),II,I1
980   FORMAT (14F7.5,5X,I2,I3)
      I1 = I1 + 1
      WRITE (13,980) (SRATES(L),L=15,28),II,I1
      I1 = I1 + 1
      WRITE (13,980) (SRATES(L),L=29,42),II,I1
```

```
      I1 = I1 + 1
      WRITE (13,980) (SRATES(L),L=43,56),II,I1
      I1 = I1 + 1
      WRITE (13,985) (SRATES(L),L=57,66),II,I1
985   FORMAT (10F7.5,25X,I2,I3)
      I1 = 20* I
      WRITE (13,980) (SRATES(L),L=67,80),II,I1
      I1 = I1 + 1
      WRITE (13,988) (SRATES(L),L=81,87),II,I1
988   FORMAT (7F7.5,40X,I2,I3)
370   CONTINUE
385   I1 = 2*I-1
      I2 = 2*I
      WRITE (13,950) (SRATEC(I,L),L=1,14),II,I1,
     1  (SRATEC(I,LL),LL=15,19),II,I2
950   FORMAT (14I5,5X,I2,I2,1X,/,5I5,50X,2I2,1X)
390   CONTINUE
400   CONTINUE
      ENDFILE 13
      REWIND 13
      STOP
      END
      SUBROUTINE TITLE (I,IRACES)
C     PRINT SEX AND RACE ON REPORT
901   FORMAT(' WHITE MALES',//)
902   FORMAT(' WHITE FEMALES',//)
903   FORMAT(' NONWHITE MALES',//)
904   FORMAT(' NONWHITE FEMALES',//)
905   FORMAT(' MALES',//)
906   FORMAT(' FEMALES',//)
      IF(IRACES .EQ. 1) GO TO 100
      IF(I .EQ. 1) WRITE(6,901)
      IF(I .EQ. 2) WRITE(6,902)
      IF(I .EQ. 3) WRITE(6,903)
      IF(I .EQ. 4) WRITE(6,904)
      RETURN
100   CONTINUE
      IF(I .EQ. 1) WRITE(6,905)
      IF(I .EQ. 2) WRITE(6,906)
      RETURN
      END
      SUBROUTINE LIFE(M,SRATE1,CAPL)
C
CC    PROGRAM TO CONSTRUCT AN ABRIDGED LIFE TABLE USING THE METHOD
CC    SUGGESTED IN (AN INTRODUCTION TO STOCHASTIC PROCESSES IN
CC    BIOSTATISTICS( BY  DR. C.L. CHIANG, CHAPTER 9, PAGES 203 -208
CC
CC
CC    WRITTEN BY HOLLY HOLLINGSHEAD , APRIL,1971 FOR SANTA CLARA COUNTY
CC    PLANNING DEPARTMENT, RESEARCH SECTION
CC
C
      REAL QHAT(19),D(19),N(18),M(19),A(18),CAPL(19),L(19),E(19)
      REAL T(19),SRATE1(19)
      INTEGER LL(19),DD(19),CAPLL(19),TT(19)
C
CC
C
```

```
      DIMENSION AGE(38)
13    FORMAT (5X,2A3,5X,F9.7,5X,I6,5X,I5,5X,F4.2,5X,I6,5X,I7,5X,
     X    F5.2,5X,F8.6,5X,F8.6)
   14 FORMAT (5X, 2A3, 5X, F9.7,5X, I6, 5X, I5, 5X,      9X,I6,5X,I7,5X,
     X    F5.2,5X,F8.6,5X,F8.6)
16    FORMAT (6X,'AGE',8X,'% DYING',6X,'NUMBER',4X,'NUMBER',
     X3X,'INTERVAL',3X,'NUMBER',4X,'YEARS OF',4X,'YEARS',5X,
     X'SURVIVAL',/,18X,'DURING',4X,'SURVIVING',2X,'DYING IN',
     X4X,'% OF',4X,'OF YEARS',3X,'ALL LIVES',3X,'OF LIFE',6X,
     X'RATE',/,17X,'INTERVAL',4X,'INTERVAL',3X,'INTERVAL',3X,
     X'LIFE',5X,'LIVED',4X,'REMAINING',2X,'REMAINING')
C
C     READ IN N(I), LENGTH OF THE I-TH INTERVAL
      DATA N/1,4,16*5/
C     READ IN A(I),FRACTION OF LAST AGE INTERVAL OF LIFE
      DATA A/.10,.39,.46,.54,.57,.49,.50,.52,.54,.54,.54,.53,.52,
     1       .52,.52,.51,.51,.45/
C     READ IN AGE CATEGORY LABELS
      DATA AGE/3H  0,3H-1 ,3H  1,3H-4 ,3H  5,3H-9 ,3H 10,3H-14,
     1         3H 15,3H-19,3H 20,3H-24,3H 25,3H-29,3H 30,3H-34,
     2         3H 35,3H-39,3H 40,3H-44,3H 45,3H-49,3H 50,3H-54,
     3         3H 55,3H-59,3H 60,3H-64,3H 65,3H-69,3H 70,3H-74,
     4         3H 75,3H-79,3H 80,3H-84,3H 85,3H-+  /
C
      DO 99 I= 1,19
      E(I) = 0.
       (I) = 0.
99    CONTINUE
C
C     CALCULATE QHAT(I),PROPORTION DYING DURING INTERVAL . FOR
C     DERIVATIONS OF FORMULAE ,PLEASE SEE ABOVE -CITED REFERENCE.
C
C
C
      DO 50 I=1,18
      QHAT(I) = (N(I)* M(I))/(1.0 + (1.0 - A(I)) * N(I)* M(I))
   50 CONTINUE
      QHAT(19) = 1.0
C     PROPORTION DYING  IN 85+ AGE GROUP IS UNITY.
C
C     USING  THE QHAT(I), CALCULATE THE D(I),NUMBER OF DEATHS IN INTERVAL
C     X(I) TO  X(I+1), AND (L(I), NUMBER  OF PERSONS SURVIVING TO EXACT
C     AGE  MARKING BEGINNING OF  EACH AGE INTERVAL.
C
C     DUE TO THE PECULIARITIES OF FORTRAN, IT IS NOT POSSIBLE TO USE ZERO
C     FOR AN ARRAY SUBSCRIPT. THUS, THE FIRST VALUES IN EACH COLUMN OF
C     THESE LIFE TABLES WHICH WOULD NORMALLY BE SUBSCRIPTED WITH A ZERO
C     (E.G. L(0) ) WILL BE;REFERRED TO WITH SUBSCRIPT OF 1.
C
C
      L(1) = 100000.0
      D(1) = QHAT (1) * L(1)
      DO 60  I= 1,18
      L(I+1) = L(I) - D(I)
      D(I+1) = L(I+1) * QHAT (I+1)
   60 CONTINUE
C
C
C     CALCULATE CAPL(I),NUMBER OF YEARS LIVED IN INTERVAL X(I) TO X(I+1)
C     BY THE L(I) SURVIVORS AT AGE  X(I)| CAPL (I) ALSO GIVES STATIONARY
```

```
C      POPULATION VALUES.
C
       DO 65  I= 1,18
       CAPL(I) =  N(I) * ( L(I)-D(I)) + A(I) * N(I) * D(I)
    65 CONTINUE
       CAPL(19) = L(19)/M(19)
C
C
C      CALCULATE T(I), TOTAL NUMBER OF YEARS REMAINING TO ALL PEOPLE
C      ATTAINING AGE X(I)
C
       DO 70 I=1,19
       DO 68 J=I,19
       T(I)  =  T(I) +CAPL(J)
    68 CONTINUE
    70 CONTINUE
C
C
C      CALCULATE  E(I),EXPECTATION OF LIFE (AVERAGE  YEARS OF LIFE REMAINING)
C
       DO 75 I=1,19
       E(I) = T(I) / L(I)
    75 CONTINUE
C
C
C      CALCULATE  S-RATES (PROBABILITY OF SURVIVING  FROM ONE AGE CATEGORY
C      TO THE NEXT).
C
C
C
C      USING METHOD IN PREVIOUS LIFE TABLE COMPUTATIONS FOR PLANNING DEPT.
C
       SRATE1 (1) =  CAPL(1)/100000.0
       SRATE1 (2) =  CAPL(2)/(CAPL(1)*4.)
       SRATE1 (3) =  CAPL(3) / (CAPL(1)+CAPL(2))
       SRATE1 (19) =  CAPL(19) / (CAPL (18) + CAPL(19))
       DO 85  I= 4,18
       SRATE1(I) = CAPL (I) / CAPL(I-1)
    85 CONTINUE
C
C
C      PRINT LIFE  TABLES
C
C
C      CONVERT CERTAIN VALUES TO INTEGER FORM FOR PRINTING
       DO 90 I=1,19
       LL(I) =     (L(I)+.5)
       CAPLL(I) =      (CAPL(I) +.5)
       DD(I) =      (D(I) +.5)
       TT(I) =      (T(I) +.5)
    90 CONTINUE
C
C      PRINT VALUES
C      PRINT HEADERS
       WRITE (6,16)
       DO  100  I =1,18
       WRITE (6,13) AGE(2*I-1),AGE(2*I),QHAT(I),LL(I),DD(I),A(I),CAPLL(I)
      1,TT(I),E(I),SRATE1(I)
   100 CONTINUE
       WRITE (6,14)AGE(37),AGE(38),QHAT(19),LL(19),DD(19),CAPLL(19),TT(19
```

```
     1),E(19),SRATE1(19)
C
C
C
C
      RETURN
      END
```

GROUPT: Grouping Analysis

GROUPT is written in standard Fortran IV. Since the program is "compute bound" the program should be compiled under Fortran H level, and used on the fastest machine possible.

Input devices are 5 for cards, 8 for the data scores, and 9 for the contiguity matrix. Output is both 6 for printout and 10 for the clustering steps. Core requirements presently are 248K to handle 1,200 OTU's. CPU time increases linearly with the number of tracts.

SYSIN Input

Card 1.	Columns 1-80	any title
Card 2	Columns 1-4	integer number of tracts
	5-8	integer number of dimensions
	9-16	integer maximum of contiguous tracts
	17-20	1= echo printout of scores and contiguity matrix
		0= no echo (default)
Card 3	Columns 1-80	Format of score input on unit 8

SCORE Input

Input of tract score data is from device 8, in the format of Card 3 above, and must be fixed/blocked form.

Contiguity Input

The matrix of contiguous tract numbers is read from device 9 in fixed/blocked form. The format is ten integer fields of six columns each (10I6):

Columns	1-6	tract identification number
	7-12	first contiguous tract identification
	13-18	second " " "
	19-24	third " " "
	25-30	4th " " "
	31-36	5th " " "
	37-42	6th " " "
	43-48	7th " " "
	49-54	8th " " "
	55-60	9th " " "

Current restrictions are: 1,200 tracts, 12 dimensions and a maximum of 9 contiguous tracts. Array sizes and common blocks must be changed if any of these limits are violated.

Description

MAIN - reads title, control parameters, and format of score input, writes out the information and calls the subroutines.

BLOCK DATA - initializes labeled common blocks.

SUBROUTINE DATAIN - reads score and contiguity data.

SUBROUTINE CLUS - performs the actual regionalization algorithm. Starting with N clusters each containing a single tract and numbered according to their input order, those two clusters which are contiguous and most similar are combined and the resulting cluster labeled with the lesser of the two codes. This cycle is repeated N-1 times. The operation is controlled by the contiguity matrix LOCAT. The tracts joined, join value, and change in value from the previous step are listed for each cycle.

SUBROUTINE OBJ and UPDATE - OBJ evaluates the clustering criteria and UPDATE updates the vectors at the end of each cycle. Ward's error sum of the squares is the objective function used. In the multivariate form it is equivalent to minimizing.

$$\sum_{j}^{m}\left(\sum^{g} x_j - \left(\sum^{g} x_j\right)^2 \Big/ g\right)$$ for each cycle; where x_j is a vector of scores for all g groups across j dimensions. UPDATE increments g and the vectors of sums and sums of squares for those tracts that now form cluster (i,k).

JCL Needed

```
// JOB                                                    (1)
//JOBLIB DD DSN=WYL.AB.PML.PROGLIB,DISP=SHR               (2)
//PROCLIB DD DSN=WYL.AB.PML.PROCLIB,DISP=SHR              (3)
// EXEC GROUPT,REGION=248K                                (4)
//FT09F001 DD DSN=WYL.AB.PML.IDEX,DISP=OLD                (5)
//FT08F001 DD DSN=WYL.AB.PML.RVARS,DISP=OLD               (6)
//FT10F001 DD DSN=WYL.AB.PML.TEST6,DISP=(,CATLG),         (7)
// UNIT=3330,VOL=SER=MTBAG1,SPACE=(TRK,(2,2),RLSE),       (8)
// DCB=(LRECL=80,BLKSIZE=3200,RECFM=FB)                   (9)
//SYSIN DD *                                             (10)
```

Sample output follows the source listing.

Costs

For two test runs, the parameters were:

	RUN I	RUN II
OTU's	440	1042
Neighbors	5	9
Steps	364	964
CPU seconds	4.68	9.67
IO	92	263
Costs	$10.18	$15.78

```
C
C.... PROGRAM  GROUPT
C
C.....      A CONTIGUITY CONSTRAINED GROUPING ALGORITHM FOR LARGE
C.....      SPATIAL DATA SETS
C.....      BY PHILIP LANKFORD
C.....         RESEARCH PLANNER
C.....         ASSOCIATION  OF BAY AREA GOVERNMENTS
C.....         BERKELEY, CA 94705
C
C.....      VERSION 1.0     APRIL,1976
C
C
C  EXECUTIVE ROUTINE
       COMMON/INIDIC/N,M,KCON,NCL
       COMMON/FIN/FMT(20)
       DIMENSION TITL(20)
       WRITE(6,40)
    40 FORMAT('1'//20X'PROGRAM GROUPT'//20X,'A CONTIGUITY CONSTRAINED GROU
      *PING ALGORITHM FOR LARGE SPATIAL DATA SETS'//20X,'BY PHILIP LANKFO
      *RD'//20X,'ABAG'////20X,'VERSION 1.0, APRIL 1976')
C  READ TITLE CARD
       READ(5,1,END=500) TITL
     1 FORMAT(20A4)
C  READ BASIC CONTROL CARD
C  N = NUMBER OF TRACTS
C  M = NUMBER OF DIMENSIONS
C  KCON = MAXIMUM NUMBER OF CONTIGUOUS TRACTS
C  NCL = ECHO PRINT CONTROL
C
       READ(5,2) N,M,KCON,NCL
     2 FORMAT(4I4)
C
C  READ FORMAT OF SCORE INPUT
       READ(5,1) FMT
C
       WRITE(6,3) TITL,N,M,KCON,NCL,FMT
     3 FORMAT('0',132('-')/'0',20A4//20X'NUMBER OF TRACTS TO BE GROUPED',
      *T60,I5//20X,'NUMBER OF VARIABLES',T60,I5//20X,'MAXIMUM NUMBER OF C
      *ONTIGUOUS TRACTS',T60,I5//20X,'CONTIGUITY LISTING',T60,I5//20X,
      *'FORMAT OF SCORE INPUT: ',20A4)
C
       CALL DATIN
       CALL CLUS
500    STOP
       END
       BLOCK DATA
       COMMON/CONTIG/LOCAT(1100,9),INUM(1100)
       DATA INUM/1100*1/,LOCAT/9900*0/
       END
```

```
C
      SUBROUTINE DATIN
      COMMON/CONTIG/LOCAT(1100,9),INUM(1100)
      COMMON/DATUM/WSUM(1100,12),WSSUM(1100,12)
      COMMON/INIDIC/N,M,KCON,NCL
      COMMON/FIN/FMT(20)
      DIMENSION D(12),ITR(1100)
C READ DATA - FACTOR SCORES
C FOR EACH TRACT READ DATA, WRITE OUT DATA ACCORDING TO NCL
C
      DO 100 I=1,N
      READ(8,FMT) (D(J),J=1,M)
      IF(NCL.GT.0) WRITE(6,1) I,(D(J),J=1,M)
1     FORMAT(1X,I7,12G10.3)
C SET DATA ARRAYS
      DO 100 J=1,M
C FACTOR SCORE DATA, WSUM(TRACT I, DIMENSION J)
      WSUM(I,J)=D(J)
C FACTOR SCORE SQUARED
  100 WSSUM(I,J)=D(J)**2
C
C READ CONTIGUITY MATRIX
      DO 200 I=1,N
  200 READ(9,2) ITR(I),(LOCAT(I,J),J=1,KCON)
    2 FORMAT(10I6)
C REMOVE RECIPROCAL PAIRS FROM CONTIGUITY MATRIX
      DO 300 I=2,N
      DO 300 J=2,KCON
  300 IF(LOCAT(I,J).LE.I) LOCAT(I,J)=0
      IF(NCL.EQ.0) RETURN
C
      DO 400 I=1,N
  400 WRITE(6,3) ITR(I), (LOCAT(I,J),J=1,KCON)
    3 FORMAT(' TRACT',I7,20X,9I7)
C
      RETURN
      END
C
      SUBROUTINE CLUS
      LOGICAL BO(1100)/1100*.FALSE./
      INTEGER IV(1100)/1100*999/,J2/-9999/,I22/0/
      COMMON/INIDIC/N,M,KCON,NCL
      COMMON/DATUM/WSUM(1100,12),WSSUM(1100,12)
      COMMON/CONTIG/LOCAT(1100,9),INUM(1100)
      REAL ESS(1100,9)/9900*0.0/,DIF/0.0/,XDD/1.0E73/,XDX/1.0E73/
      WRITE(6,1)
    1 FORMAT('0  CLUSTERING CRITERIA - WARDS ERROR-SUM-OF-SQUARES OBJECT
     *IVE FUNCTION'//' CYCLE ',T9,'TRACT 1',T17,'TRACT 2',T25,' CRITERIA
     * VALUE',T40,' DIFFERENCE'/' NUMBER ',//T4,' (1) ',T12,' (2) ',T20,
     *' (3) ',T32,' (4) ',T42,' (5) '///)
      NN=N-1
```

```
C SET UP  MATRIX OF ERROR-SUMS-OF-SQUARES
      DO 300 I=1,N
      IS=0
      DO 400 J=2,KCON
      IF(LOCAT(I,J).EQ.0) GO TO 400
      KJ=LOCAT(I,J)
      ESS(I,J)=OBJ(I,KJ)
      IS=IS+1
  400 CONTINUE
  300 BO(I)=IS.EQ.0
      LAST=0
C START CLUSTERING CYCLES
      DO 2 I2=1,NN
      DO 3 I=1,N
C TEST VALUE AGAINST -9999
      IF(LOCAT(I,1).EQ.J2) GO TO 555
C TRUE WHEN A TRACT HAS NO CONTIGUOUS TRACTS
      IF(BO(I)) GO TO 3
      IF(LOCAT(I,1).EQ.I22) GO TO 666
      IS=0
      K=LOCAT(I,1)
      DO 4 J=2,KCON
      IF(LOCAT(I,J).EQ.0) GO TO 4
      KL=LOCAT(I,J)
C TEST VALUE AGAINST -9999
      IF(KL.EQ.J2) GO TO 888
      IF(KL.EQ.I22) GO TO 777
      IS=IS+1
      IF(ESS(I,J).GT.XDD) GO TO 4
      XDD=ESS(I,J)-0.0000001
      ICELL=K
      JCELL=KL
      GO TO 4
C UPDATE J'TH ELEMENT, THAT IS J2, COL 2-7
  888 IF(K.EQ.I22) GO TO 101
      LOCAT(I,J)=I22
  777 IF(IV(K).EQ.LAST) GO TO 101
      IS=IS+1
      IV(K)=LAST
      ESS(I,J)=OBJ(K,I22)
      IF(ESS(I,J).GT.XDD) GO TO 4
      XDD=ESS(I,J)
      ICELL=K
      JCELL=I22
      GO TO 4
  101 LOCAT(I,J)=0
    4 CONTINUE
      BO(I)=IS.EQ.0
      GO TO 3
```

```
C UPDATE I'TH ROW, THAT IS J2, IN COL 1
C SET TO ZERO
  555 LOCAT(I,1)=I22
C TRUE WHEN A TRACT HAS NO CONTIGUOUS TRACTS
      IF(BO(I)) GO TO 3
  666 IS=0
      DO 6 J=2,KCON
      IF(LOCAT(I,J).EQ.0) GO TO 6
      KJ=LOCAT(I,J)
      IF(IV(KJ).EQ.LAST) GO TO 66
      IV(KJ)=LAST
      IS=IS+1
      ESS(I,J)=OBJ(I22,KJ)
      IF(ESS(I,J).GT.XDD) GO TO 6
      XDD=ESS(I,J)
      ICELL=I22
      JCELL=KJ
      GO TO 6
   66 LOCAT(I,J)=0
    6 CONTINUE
      BO(I)=IS.EQ.0
    3 CONTINUE
C HERE AT END OF GROUPING PROCESS
      IF(XDX.EQ.XDD) GO TO 200
      IV(ICELL)=I2
      IV(JCELL)=I2
      LAST=I2
      DIF=XDD-DIF
      WRITE(6,100) I2,ICELL,JCELL,XDD,DIF
  100 FORMAT(' ',I5,2I8,2F20.1)
      WRITE(10,100) I2,ICELL,JCELL,XDD,DIF
      DIF=XDD
C UPDATE DATA STORES
      CALL UPDATE(ICELL,JCELL)
      J2=JCELL
      I22=ICELL
C UPDATE ROW ICELL OF ESS
      XDD=1.0E73
    2 CONTINUE
C
C
200   ENDFILE 10
      JZZ=J2+1
      WRITE(6,201) JZZ
  201 FORMAT('0',132('*')//T20,'AT CYCLE',I8//T20,'NO FURTHER GROUPING O
     *F TRACTS WAS POSSIBLE UNDER CONTIGUITY USED')
      WRITE(6,202)
  202 FORMAT('0 ',T20,' EXPLANATION OF OUTPUT ***'//T30, 'COLUMN..(1) ',
     *T50,' CYCLE NUMBER,FROM 1 TO N-1'//T30, 'COLUMN..(2) AND (3)
     *',T50,' SEQUENCE CODE OF TRACT THAT WAS MOST SIMILAR TO TRACT GIVE
     *N IN COL 3'//T30,'COLUMN..(4)',T50,' PRESENT VALUE OF CLUSTERING C
     *RITERIA'//T30,'COLUMN..(5)',T50,' CHANGE IN VALUE SINCE PREVIOUS C
     *YCLE'//'0',22('.'),' END OF PROGRAM ',22('.'))
      RETURN
      END
```

```
      FUNCTION OBJ(I,J)
      COMMON/DATUM/WSUM(1100,12),WSSUM(1100,12)
      COMMON/INIDIC/N,M,KCON,NCL
      COMMON/CONTIG/LOCAT(1100,9),INUM(1100)
      E=0.0
      X=FLOAT(INUM(I)+INUM(J))
      DO 100 L=1,M
  100 E=E+(WSSUM(I,L)+WSSUM(J,L))-((WSUM(I,L)+WSUM(J,L))**2/X)
      OBJ=E
      RETURN
      END
      SUBROUTINE UPDATE(I,J)
      COMMON/DATUM/WSUM(1100,12),WSSUM(1100,12)
      COMMON/INIDIC/N,M,KCON,NCL
      COMMON/CONTIG/LOCAT(1100,9),INUM(1100)
      DO 100 L=1,M
      WSUM(I,L)=WSUM(I,L)+WSUM(J,L)
  100 WSSUM(I,L)=WSSUM(I,L)+WSSUM(J,L)
      INUM(I)=INUM(I)+INUM(J)
      RETURN
      END
```

GROUPM: Defining Social Areas

GROUPM reads the grouping information from a GROUPT run and produces for a given step, group memberships and means on each dimension.

INPUT

Card 1	Format (3I4)	
Columns	1- 4	GMAX, step number cutoff for analysis
	5- 8	NUMOTU, total number of observations
	9-12	NUMDIM, number of dimensions
Card 2	Format (20A4)	
Columns	1-80	PMT, variable format of raw data

File FT10F001 contains the GROUPT output dataset of the step information formatted as (6X,2I8).

File FT20F001 is the raw data read according to the second card format. The read statement seeks an identification sequence number and then the data for the observation, up to 12 dimensions.

The program uses 126K.

```
C  STARTING WITH 1100 CLUSTERS EACH CONTAINING A SINGLE TRACT
C  AND NUMBERED IN ORDER, THOSE TWO CLUSTERS WHICH HAVE AT
C  LEAST ONE NEIGHBOR IN COMMON AND MOST SIMILAR ARE COMBINED
C  AND THE RESULTING UNION CLUSTER IS LABELLED WITH THE LESSER
C  OF THE TWO CODES. THIS IS REPEATED GMAX-1 TIMES.
C   ALLOWS TWELVE DIMENSIONS
C  PRODUCES GROUP MEMBER  AND ISOLATES LIST FOR A GIVEN STEP
       DIMENSION G(1100),IND(1100,2),ISO(1100),GROUP( 1100),
      *ID(1100),GR(1100)
       DIMENSION D(1100,12),DIM(12),FMT(20)
       INTEGER GMAX,SEARCH,G,GROUP,GR
       DATA ISO/1100*0/
C  STEP NUMBER
       READ(5,3) GMAX,NUMOTU,NUMDIM,FMT
3      FORMAT(3I4/20A4)
       WRITE(6,8) GMAX,NUMOTU,NUMDIM
8      FORMAT('1STEP = ',I5/'0NUMBER OF TRACTS = ',I5/
      *'0NUMBER OF DIMENSIONS = ',I5)
C  READ VECTOR
       DO 50 I=1,GMAX
       READ(10,1) (IND(I,J),J=1,2)
1      FORMAT(6X,2I8)
       DO 50 J=1,2
50     IF(IND(I,J).GT.NUMOTU) IND(I,J)=0
.      DO 60 I=1,NUMOTU
60     READ(20,FMT) ID(I),(D(I,J),J=1,NUMDIM)
C
       IG=1
       DO 500 I=1,GMAX
       SEARCH=IND(I,1)
       IF(SEARCH.LT.1) GO TO 500
       G(1)=SEARCH
       G(2)=IND(I,2)
       IND(I,1)=0
       IND(I,2)=0
       N=1
       K=3
C  SCAN REMAINDER OF LIST
       L=I+1
99     DO 100 J=L,GMAX
       IF(IND(J,1).EQ.SEARCH) GO TO 125
       IF(IND(J,2).EQ.SEARCH) GO TO 150
       GO TO 100
C  HERE FOR SIMPLE ADDITION OF A MEMBER
125    G(K)=IND(J,2)
C  UPDATE
155    IND(J,1)=0
       IND(J,2)=0
       K=K+1
       GO TO 100
C  HERE FOR ADDITION OF ANOTHER MEMBER
150    CONTINUE
       G(K)=IND(J,1)
       GO TO 155
```

```
C
100   CONTINUE
      N=N+1
      IF(N.GE.K) GO TO 98
      SEARCH=G(N)
      GO TO 99
C
98    K=K-1
      DO 223 J=1,K
      LLL=G(J)
      GROUP( J)=ID(LLL)
223   CONTINUE
      DO 299 J=1,K
299   GR(J)=9999999
      J=1
C SORT SECTION * * * * * * * * * * * * * * * * * * * * * * * * * *
305   DO 300 KK=1,K
300   GR(J)=MINO(GROUP(KK),GR(J))
      DO 301 KK=1,K
301   IF(GR(J).EQ.GROUP(KK)) GROUP(KK)=9999999
      IF(J.EQ.K) GO TO 310
      J=J+1
      GO TO 305
C
310   WRITE(6,4) IG,K,(GR(J),J=1,K)
4     FORMAT('0GROUP ',I8,' HAS ',I8,' MEMBERS : ',/(1X,10I8/))
      DO 224 NN=1,NUMDIM
224   DIM(NN)=0.
      DO 225 J=1,K
      M=G(J)
      DO 226 NN=1,NUMDIM
226   DIM(NN)=DIM(NN)+D(M,NN)
225   ISO(M)=1
      DO 227 NN=1,NUMDIM
227   DIM(NN)=DIM(NN)/FLOAT(K)
      WRITE(6,10) (DIM(NN),NN=1,NUMDIM)
10    FORMAT('0GROUP AVERAGES :'6F15.4/1X,6F15.4)
      IG=IG+1
C
500    CONTINUE
C
      KISO=0
      DO 525 I=1,NUMOTU
      IF(ISO(I).EQ.0) KISO=KISO+1
525   IF(ISO(I).EQ.0) GROUP(KISO)=ID(I)
      WRITE(6,2) KISO,(GROUP(I),I=1,KISO)
2     FORMAT('0THERE ARE ',I6,' ISOLATES : '/(1X,10I8/))
      STOP
      END
```

ADAM Program

ADAM is the areal demographic allocation model. It requires four input files, and writes the small area population projections to a fifth file.

Input Description

From device 10 the program read the base year age/sex distribution for 28 cohorts for each of the 96 social areas. There are two "cards" per social area; males and females. For each card the first six columns are blank, followed by 14 fields of six columns for each of the proportions. The format for each social area is: (6X, 14F6.4/6X,14F6.4).

From device 20 the program obtains the 4 cycles of 28 cohort totals. These totals, produced by APPLE, are the regional cohort control totals. There are four "cards" per projection cycle, beginning with the base year. Each card contains seven cohort totals in fields of ten, males followed by females. The format is: (7 F10.0).

From device 30 the three cohort change vectors are read. For each change vector, the 28 cohorts are read in 14 fields of six columns, males followed by females. The format is: (14F6.4/14F6.4).

There are a total of six "cards" required.

The fourth file, read from device 40, contains the four cycles of social area total populations. Beginning with the base year, and for each of the three projection cycles, the file contains 96 "cards" containing the population total in columns 1-8. A total of 384 "cards" are required. Format (F8.0).

JCL Required

Sample cards needed to run ADAM follow the Subroutine PRINT listing.

```
C
C
C     --A D A M--
C
C     AREAL DEMOGRAPHIC ALLOCATION MODEL
C
C     DEVELOPED APRIL 1977
C
C     CASTRO,LANKFORD,SHYER
C
C
      REAL DISTN,HHPOP,HHCHNG,SAPOP,SAPAS,COHORT
      COMMON/NANCY/ SAPAS,HHPOP,SAPOP
      DIMENSION DISTN(96,28),HHPOP(4,28),HHCHNG(3,28),
     *SAPOP(96,4),SAPAS(4,96,28),COHORT(28),TPOP(96),FACT(4)
      DATA TPOP/96*0./,FACT/1.0,1.0154,1.0150,1.0146/
C
C     READ IN 1970 AGE/SEX DISTRIBUTIONS BY SOCIAL AREA
C
      DO 1 I=1,96
1     READ (10,100) (DISTN(I,J),J=1,28)
100   FORMAT (6X,14F6.4,/,6X,14F6.4)
C
C     READ IN 4 CYCLES OF REGIONAL COHORT TOTALS
C
      DO 2 K=1,4
2     READ (20,200) (HHPOP(K,J),J=1,28)
200   FORMAT (7F10.0,10X)
C
C     READ IN 3 CHANGE VECTORS FOR EACH OF THE 28 COHORTS
C
      DO 3 K=1,3
3     READ (30,300) (HHCHNG(K,J),J=1,28)
300   FORMAT (14F6.4,/,14F6.4)
C
C     READ IN 4 CYCLES OF SOCIAL AREA POPULATIONS
C
      DO 4 K=1,4
      READ (40,400) (SAPOP(I,K),I=1,96)
400   FORMAT (F8.0,72X)
C
C     CORRECT SOCIAL AREA TOTAL POPULATIONS TO THE REGIONAL
C     COHORTS' TOTAL
C
      DO 401 I=1,96
401   SAPOP(I,K)=SAPOP(I,K)*FACT(K)
4     CONTINUE
C
C     COMPUTE 1970 SOCIAL AREA POPULATION BY AGE AND SEX FOR
C     EACH SOCIAL AREA
C
      DO 1000 I=1,96
      DO 1010 J=1,28
      COHORT(J)=0.
1010  SAPAS(1,I,J)=DISTN(I,J)*SAPOP(I,1)
1000  CONTINUE
C
C     CHECK FOR ROUNDING OR TRUNCATION ERRORS
C
```

```
      CALL ROUND(1,28,96)
502   FORMAT (I6,14F8.0,/,6X,14F8.0)
C
C     WRITE FIRST CYCLE ROUNDED RESULTS TO OUTPUT FILE
C
      DO 6 I=1,96
6     WRITE (50,500)I,(SAPAS(1,I,J),J=1,28)
500   FORMAT (I6,14F6.0,/,6X,14F6.0)
C
C     COMPUTE DISTRIBUTIONS FOR THE PROJECTIONS CYCLES
C
3000  DO 2000 K=1,3
      KL=K+1
      DO 2010 I=1,96
      DO 2020 J=1,28
C
C     IF SOCIAL AREA'S POPULATION IS BELOW THE THRESHOLD,
C     THE DISTRIBUTION IS HEAL CONSTANT
C
      IF (SAPOP(I,K).LE.6000.) GO TO 2021
      SAPAS(K+1,I,J)=(DISTN(I,J)*SAPOP(I,K)) * HHCHNG(K,J)
      GO TO 2022
2021  SAPAS(K+1,I,J)=DISTN(I,J)*SAPOP(I,K+1)
C
C     COMPUTE A RUNNING TOTAL OF RESULTING POPULATION FIGURES
C
2022  TPOP(I) = TPOP(I) + SAPAS(K+1,I,J)
2020  CONTINUE
C
C     CONTROL THE COMPUTED POPULATION AGE/SEX DISTRIBUTION TO
C     THE LAND USE MODEL'S SOCIAL AREA TOTAL
C
      DO 2012 J=1,28
      SAPAS(K+1,I,J)=(SAPAS(K+1,I,J)/TPOP(I))*SAPOP(I,K+1)
      COHORT(J)=0.
2012  CONTINUE
      WRITE (6,502) I,(SAPAS(K+1,I,J),J=1,28)
      TPOP(I) = 0.
2010  CONTINUE
C
C     CHECK FOR ROUNDING OR TRUNCATION ERRORS
C
      CALL ROUND(K+1,28,96)
      DO 2014 I=1,96
2014  WRITE (6,502)I,(SAPAS(K+1,I,J),J=1,28)
C
C     CHECK FOR BIAS IN COHORT DISTRIBUTION
C     COMPARE COMPUTED SOCIAL AREA COHORT TOTAL TO REGIONAL
C     DEMOGRAPHIC COHORT TOTAL
C
      DO 2015 J=1,28
      DO 2013 I=1,96
2013  COHORT(J)=COHORT(J)+SAPAS(K+1,I,J)
      CODIF=HHPOP(K+1,J)-COHORT(J)
2015  WRITE (6,503)KL,CODIF
503   FORMAT (I5,' CODIF=',F10.0)
C
C     COMPUTE A NEW AGE/SEX DISTRIBUTION
C
      DO 2030 I=1,96
```

```
      DO 2040 J=1,28
      DISTN(I,J)=(SAPAS(K+1,I,J))/(SAPOP(I,K+1)+.00005)
2040  CONTINUE
C
C     WRITE CYCLE RESULTS TO DISK FILE
C
      WRITE (50,500)I,(SAPAS(K+1,I,J),J=1,28)
2030  CONTINUE
2000  CONTINUE
      REWIND 50
      CALL PRINT
      STOP
      END
      SUBROUTINE ROUND(IYR,IDIM,JDIM)
      COMMON/NANCY/ SAPAS,HHPOP,SAPOP
      REAL MAT
      DIMENSION MAT(96),SAPAS(4,96,28),HHPOP(4,28),SAPOP(96,4)
C     CONTROL FOR JDIM NUMBER OF ROUNDINGS
1     DO 500 ITER=1,JDIM
      IF (IDIM.LE.28) GO TO 10
C     ROUNDING FOR COHORT TOTAL
      DO 11 I=1,IDIM
11    MAT(I) = SAPAS(IYR,I,ITER)
      GO TO 12
10    CONTINUE
C     ROUNDING FOR SOCIAL AREA TOTAL
      DO 13 I=1,IDIM
13    MAT(I) = SAPAS(IYR,ITER,I)
12    CONTINUE
C     ROUND MATRIC MAT TO APPROPRIATE CONTROL
C     IF ROUNDING TO SOCIAL AREA, TOTAL = CYCLE SOCIAL AREA POP
      IF (IDIM.EQ.28) TOT=SAPOP(ITER,IYR)
C     IF ROUNDING TO REGIONAL COHORT, TOTAL = CYCLE COHORT POP
      IF (IDIM.EQ.96) TOT = HHPOP(IYR,ITER)
C     CHECK FOR ROUND-OFF TRUNCATION ERRORS
      IDIM2=IDIM/2
      KNT = 0
398   ICK = 0
      KNT = KNT + 1
      IF (KNT.GT.4) GO TO 300
      DO 310 I=1,IDIM
C     ICK IS THE NEW TOTAL FOR THE DIMENSION TO BE TESTED
310   ICK = ICK + MAT(I)
      X1= TOT
C     COMPUTE DIFFERENCE AS INTEGER
      IDIFF = IFIX(X1)-ICK
      IF (IDIFF.EQ.0) GO TO 300
      X1 = IDIFF/IDIM
C     SAV INCREMENT AS INTEGER
      L=IABS(IFIX(X1))
C     COMPUTE RESIDUAL
      IB=IABS(MOD(IDIFF,IDIM))
C     DISTRIBUTE DIFF GE IDIM
      IX=ISIGN(1,IDIFF)
      IF (L.EQ.0) GO TO 400
```

```
C
      DO 370 I=1,IDIM
      MAT(I)=MAT(I)+IX*L
370   IF (MAT(I).LT.0.) MAT(I)=0.
C
C
C     NOW DISTRIBUTE THE BALANCE
400   L=1
      IF (IB.EQ.0) GO TO 399
      IF (IB.LT.IDIM2) GO TO 380
C     DISTRIBUTE TO BOTTOM HALF FIRST
      DO 381 I=IDIM2,IB
      MAT(I)=MAT(I)+(IX*L*(MAT(I)/ICK))
381   IF (MAT(I).LT.0.) MAT(I)=0.
C     DISTRIBUTE BALANCE TO TOP HALF
      IB=IDIM2
380   DO 382 I=1,IB
      MAT(I)=MAT(I)+(IX*L*(MAT(I)/ICK))
382   IF (MAT(I).LT.0.) MAT(I)=0.
399   GO TO 398
C
300   CONTINUE
C     CORRECT THE SAPAS MATRIX WITH THE NEWLY COMPUTED VALUES
      IF (IDIM.EQ.28) GO TO 51
      DO 501 I=1,96
501   SAPAS(IYR,I,ITER) = MAT(I)
      GO TO 52
51    CONTINUE
      DO 504 I=1,28
504   SAPAS(IYR,ITER,I) = MAT(I)
52    CONTINUE
500   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE PRINT
      REAL*8 AGE
      REAL MTOT
      DIMENSION SAPAS(4,96,28),TOT(4),CTOT(14),MTOT(4),FTOT(4),
     *AGE(15),IPER(4)
      DATA IPER/1970,1975,1980,1985/,MTOT/4*0./,FTOT/4*0./,CTOT/14*0./,
     *TOT/4*0./,
     *AGE/'  0-4 ','   5-9  ','  10-14 ','  15-19 ','  20-24 ',
     *'  25-29 ','  30-34 ',
     *'  35-39 ','  40-44 ','  45-49 ','  50-54 ','  55-59 ',
     *'  60-64 ','   65+  ','  TOTAL '/
      DO 1000 K=1,4
      DO 1000 I=1,96
1000  READ (50,100) (SAPAS(K,I,J),J=1,28)
100   FORMAT (6X,14F6.0,/,6X,14F6.0)
      DO 2000 I=1,96
      WRITE (6,101)
101   FORMAT ('1',49X,'ADJUSTED AGE AND SEX DISTRIBUTION',/)
      DO 2010 K=1,4
      WRITE (6,102) I,IPER(K)
102   FORMAT (/,' ',58X,'SOCIAL AREA',I3,/,63XI5,//)
      WRITE (6,103)AGE
103   FORMAT (11X,15A8)
      DO 2011 J=1,28
2011  TOT(K) = TOT(K) + SAPAS(K,I,J)
      DO 2020 J=1,14
      CTOT(J)=CTOT(J) + SAPAS(K,I,J) + SAPAS(K,I,J+14)
      MTOT(K) = MTOT(K) + SAPAS(K,I,J)
      FTOT(K) = FTOT(K) + SAPAS(K,I,J+14)
2020  CONTINUE
      WRITE (6,350) (SAPAS(K,I,J),J=1,14),MTOT(K)
      WRITE (6,301) (SAPAS(K,I,J),J=15,28),FTOT(K)
      WRITE (6,302) (CTOT(J),J=1,14),TOT(K)
      MTOT(K) = 0.
      FTOT(K) = 0.
      TOT(K) = 0.
      DO 2012 J=1,14
2012  CTOT(J) = 0.
350   FORMAT (' MALES',4X,15F8.0)
301   FORMAT (' FEMALES',2X,15F8.0)
302   FORMAT (' TOTAL',4X,15F8.0)
2010  CONTINUE
2000  CONTINUE
      RETURN
      END
/*
//GO.FT10F001 DD DSN=WYL.AB.NAG.SADIST,UNIT=3330,DISP=OLD
//GO.FT20F001 DD DSN=WYL.AB.ACC.XHHPOP,DISP=OLD,UNIT=3330
//GO.FT30F001 DD DSN=WYL.AB.ACC.CHVECT1,DISP=OLD,UNIT=3330
//GO.FT40F001 DD DSN=WYL.AB.NAG.XSAPOP,UNIT=3330,DISP=OLD
//GO.FT50F001 DD DSN=WYL.AB.ACC.FRIDAY13,UNIT=3330,DISP=(,CATLG),
//   VOL=SER=MTBAG1,DCB=(LRECL=90,RECFM=FB,BLKSIZE=2700),
//   SPACE=(TRK,(2,3),RLSE)
/*
//
```